A New Monte Carlo Method for Quantum Spin Systems : Technical Details of the Method

KADOWAKI, Shinji; UEDA, Akira

KADOWAKI, Shinji ...[et al]. A New Monte Carlo Method for Quantum Spin Systems : Technical Details of the Method. Memoirs of the Faculty of Engineering, Kyoto University 1991, 53(1): 1-18

Mem. Fac. Eng., Kyoto Univ. Vol. 53, No. 1 (1991)

A New Monte Carlo Method for Quantum Spin Systems

—*Technical Details of the Method*—

by

Shinji KADOWAKI* and Akira UEDA†

(Received August 21, 1990)

Abstract

A simple Monte Carlo method to evaluate the partition function of finite quantum spin systems, which was proposed in the previous paper[1], was applied to obtain accurate values for thermodynamic quantities down to rather low temperatures for one and two dimensional systems. In the present paper some technical details and comments on the method are given.

1. Introduction

The high temperature expansion method (HTEM)[2] gives us exact results for thermodynamic quantities at high temperatures in the sense that the first ten to twenty coefficients of the cumulant expansions are exactly evaluated. The HTEM has been applied to various lattice systems.[2] Since, however, the coefficients so far evaluated are limited to a bit higher than the twentieth term after much unremitting endeavor, it is too insufficient to examine the thermodynamic properties at lower temperatures. In order to supply this deficiency, a Monte Carlo method for the spin-1/2 quantum Heisenberg system was originally proposed by Handscomb[3] more than twenty years ago. This method is based on the high temperature expansion of the partition function, and introduces an importance-sampling method similar to the Metropolis method. Recently the practical computing scheme of Handscomb's method was improved by Lyklema.[4] Subsequently, the method was modified by Chakravarty and Stein[5] so as to be applicable to the spin-1/2 quantum XY-spin system in one dimension, and also by Lee et al[6] to the antiferromagnetic quantum Heisenberg spin systems for linear, square and triangle lattices, respectively. Furthermore, Lee et al.'s method was ap-

---

* Department of Applied Mathematis, Fuculty of Science, Okayama University of Science, Okayama 700.

† Department of Applied Mathematics Physics, Faculy of Engineering, Kyoto University, Kyoto 606.

plied to calculate the correlation length for square lattice systems by Manousakis and Salvador[7]. All of these methods use an importance-sampling technique and avoid evaluating the partition function directly. On the contrary we have developed a new approach[1] within a framework of HTEM, in which the partition function is evaluated numerically accurately as well as directly with use of the Monte Carlo method. The method is to evaluate the values of the moment-expansion coefficients up to terms of order as high as tens to hundreds of thousands and was found to yield unexpectedly precise values of thermodynamic quantities for the spin 1/2 quantum Heisenberg systems at rather low temperatures. In the present paper we will give some technical details of the method.

In the next section a brief summary of some mathematical formulas used in our method are given for the convenience of readers. The Monte Carlo (MC) sampling for the expansion coefficients is explained in §3. Especially, an algorithm for the cycle descomposition of permutation is described in detail. Subsequently, how to estimate traces is described. In §4 powerful interpolation method for evaluating many expansion coefficients is introduced. Some technical problems which arise in an application of our method are described in §5, and some discussions are made in §6.

2. Review of the required Formulas

1. Hamiltonian

We consider the spin-1/2 isotropic Heisenberg system of N spins. The Hamiltonian is given by

$$\mathcal{H}=-\frac{1}{2}J\sum_{\langle ij\rangle}\vec{\sigma}_i\cdot\vec{\sigma}_j-\mu H\sum_{i=1}^{N}\sigma_i^z. \tag{1}$$

where $\vec{\sigma}$ is the Pauli spin operator, $J(>0)$ the coupling constant, μ the magnetic moment, and H the external magnetic field in the z-direction. The summation in the first term is taken over the nearest neighbour spin pair i, j. The periodic boundary condition is assumed. Using the well-known Dirac identity[8] $\sigma_i\cdot\vec{\sigma}_j=2(i\,j)-1$ where $(i\,j)$ is the permutation operator called transposition, which interchanges the spin states at sites i and j, and shifting the zero of the energy $-(1/2)\,JN_b$, Hamiltonian (1) is rewitten as

$$\mathcal{H}=-J\sum_{b=1}^{Nb}(i_b j_b)+\mathcal{H}_0, \tag{2}$$

Here the suffix b in $(i_b\, j_b)$ is a bond-number which joints the sites i_b and j_b and the summantion $\sum_b$ is carried over the total number, N_b, of the nearest neighbour bonds. H_0 represents the second term of Eq. (1).

2. Partition function and it's high temperature expansion coefficients

The partition function can be expanded as

$$Z=\mathrm{Tr}(\exp(-\mathcal{H}/kT)) = \sum_{r=0}^{\infty} \frac{a_r}{r!}\tau^{-r} , \tag{3}$$

where k_B is the Boltzmann constant, T the temperature and τ stands for $k_B T/J$. The expansion coefficient a_r is given by

$$a_r = \sum_{C_r} A(C_r) = \sum_{C_r} \mathrm{Tr}\{P(C_r) exp(-H_0/k_B T)\} \tag{4}$$

and $P(C_r)$ can be written as

$$P(C_r) = (i_{b_1} j_{b_1})(i_{b_2} j_{b_2}) \cdots (i_{b_r} j_{b_r})(h_1) \cdots (h_s) , \tag{5}$$

where (h) is an identical operator which maps a spin state at a site h on itself and hence is omitted hereafter. C_r represents a sequence of r bond-numbers b_1, b_2, $\cdots$, b_r which are selected with repetition from a set $\{1, 2, \cdots, N_b\}$, and the summation is carried out over all possible N_b^r sequences C_r's. Any bond-number b_k uniquely corresponds to the nearest neighbour pair i_{b_k} and j_{b_k}, where a site-number i_{b_k} is an integer between 1 and N. $P(C_r)$ represented by a product of transpositions becomes a permutation of order N, because it involves all of N integers from 1 to N. According to the theory of the permutation group[9], a permutation can be decomposed into a product of independent cycles which have no common elements, as follows:

$$P(C_r) = \begin{pmatrix} 1 & 2 & 3 & \cdots & N \\ k_1 & k_2 & k_3 & \cdots & k_N \end{pmatrix} = (j_1 j_2 \cdots j_{l_1})(j_{l_1+1} \cdots j_{l_1+l_2}) \cdots (\cdots j_N) , \tag{6}$$

Where, for example, the cycle $(j_1 j_2 \cdots j_{l_1})$ stands for the cyclic permutation of the length l_1, which maps $j_2 \rightarrow j_1, j_3 \rightarrow j_2, \cdots, j_1 \rightarrow j_{l_1}$.

Now a trace $A(C_r)$ in Eq. (4) is given by

$$A(C_r) = \prod_{l=1}^{N} \{2cosh(lL)\}^{\nu_l} , \tag{7}$$

where L stands for $\mu H/k_B T$ and ν_l is the number of independent cycles of length. The ν's must satisfy

$$1\nu_1 + 2\nu_2 + \cdots + l\nu_l + \cdots + N\nu_N = N . \tag{8}$$

For zero-magnetic field Eq. (7) reduces to

$$A(C_r) = 2^{k(C_r)} , \tag{9}$$

where $k(C_r)=\sum_{l=1}^{N} \nu_l$ is the total numbei of independent cycles.

The expansion coefficients of Z, $\partial Z/\partial L$ and $\partial^2 Z/\partial L^2$ required for evaluating thermodynamic quantities are summarized as follows. For

$$Z=\sum_{r=0}^{\infty}\frac{a_r}{r!}\tau^{-r}\,, \tag{10}$$

$$a_r=\sum_{C_r}A(C_r),\; A(C_r)=\prod_{l=1}^{N}\{2cosh(lL)\}^{\nu_l} \tag{11}$$

$$\frac{\partial Z}{\partial L}=\sum_{r=0}^{\infty}\frac{b_r}{r!}\tau^{-r}\,, \tag{12}$$

for

$$b_r=\sum_{C_r}B(C_r)\,,\qquad B(C_r)=A(C_r)\sum_{l=1}^{N}l\nu_l tanh(lL) \tag{13}$$

and for

$$\frac{\partial^2 Z}{\partial L^2}=\sum_{r=0}^{\infty}\frac{dr}{r!}\tau^{-r}\,, \tag{14}$$

$$d_r=\sum_{C_r}D(C_r)\,,$$

$$D(C_r)=A(C_r)\left[\sum_{l=1}^{N}\nu_l l^2-\sum_{l=1}^{N}(l\;tanh(lL))^2+\{\sum_{l=1}^{N}l\nu_l\;tanh(lL)^2\}\right]. \tag{15}$$

In the case of the zero-magnetic field, Eq. (15) recuces to

$$D(C_r)=A(C_r)\sum_{l=1}^{N}\nu_l l^2\,. \tag{16}$$

where ν's must satisfy Eq.(8).

3. Thermodynamic quantities

The thermodynamic quantities are derived by differentiating the partition function Z. The entropy, internal energy and specific heat are given by

$$\frac{S}{Nk_B}=\frac{1}{N}lnZ+\frac{\tau}{N}\frac{1}{Z}\frac{\partial Z}{\partial\tau}\,, \tag{17}$$

$$\frac{E}{NJ}=\frac{\tau^2}{N}\frac{1}{Z}\frac{\partial Z}{\partial\tau}-\frac{1}{2}\frac{q}{2}\,, \tag{18}$$

and

$$\frac{C_H}{NJ}=\frac{\tau^2}{N}\left[\frac{1}{Z}\frac{\partial^2 Z}{\partial\tau^2}+\frac{2}{\tau}\frac{1}{Z}\frac{\partial Z}{\partial\tau}-\left(\frac{1}{Z}\frac{\partial Z}{\partial\tau}\right)^2\right], \tag{19}$$

where q is the coordination number of the lattice sites. The magnetization and susceptibility are given by

$$\frac{M}{N\mu}=\frac{1}{N}\frac{1}{Z}\frac{\partial Z}{\partial L}, \tag{20}$$

$$\frac{k_B T\chi}{N\mu^2}=\frac{1}{N}\left[\frac{1}{Z}\frac{\partial^2 Z}{\partial L^2}-\left(\frac{1}{Z}\frac{\partial Z}{\partial L}\right)^2\right], \tag{21}$$

The susceptibility in the zero-magnetic field, χ_0, is given by

$$\frac{k_B T\chi_0}{N\mu^2}=\frac{1}{N}\frac{1}{Z}\sum_{r=0}^{\infty}\frac{d_r}{r!}\tau^{-r}, \tag{22}$$

where $D(C_r)$'s in the d_r are given by Eq. (16).

3. MC sampling for the expansion coefficients

In the present paper, we consider the case of the zero-magnetic field for which the expansion coefficients a_r and d_r need to be evaluated. The procedure of the evaluation consists of two parts. Firstly, M expansion coefficients, M being to be about ten, are evaluated by a crude Monte Carlo method, of which details of the sampling are described in this section. In the second part, the rest of the expansion coefficients are evaluated by interpolation with use of a formula of the Pade type rational function, of which details are given in the next section.

3.1. MC sampling

The a_r and d_r for M specified r's, say, $r_2, r_1, \ldots, r_M$ in increasing order, are evaluated with a crude Monte Carlo method. That is, to evaluate the coefficient of the r-th term we chose r integers $b_1, b_2, \cdots, b_r$ randonly with repetition from the set $\{1, 2, \cdots, N_b\}$ and construct $P(C_r)$ as given by Eq. (5) and then compute $A(C_r)$. Repeating the procedure N_s times, we obtain the mean value, $\bar{A}_r$, of $A(C_r)$ $(\equiv a_r/N_b^r)$ and hence the MC value, $\bar{a}_r$, of a_r. They are given by

$$\bar{A}_r=\sum_{MC}A(C_r)/N_s, \tag{23a}$$

and

$$\bar{a}_r=N_b^r\bar{A}_r, \tag{23b}$$

where $\sum_{MC}$ stands for the summantion over N_s Monte Calro samples. In a similar way $\bar{D}_r$ and $\bar{d}_r$ are obtained. Computational details are given in the following subsections.

3.2. Cycle Decomposition of Permutation

Let us suppose that the N_b bonds are already numbered from 1 to N_b and the bond numbered b_k connects the nearest neighbour sites i_{b_k} and j_{b_k}. Then $P(C_r)$ as a product of the transpositions given in Eq. (5) is constructed simply by choosing r integers $b_1, b_2, \cdots, b_r$ randomly with repetition from the set $\{1, 2, \cdots, N_b\}$. On coding the computer program for the cycle decomposition of $P(C_r)$ it is convenient to use two linear arrays. One of them is an array SI for site i_{b_k}, and another one an array SJ for site j_{b_k}. The subscript of both arrays corresponds to the bond-number b_k of the spin and the numbers of spin sites i_{b_k} and j_{b_k} are stored in SI and SJ arrays, respectively. That is,

$$\mathrm{SI}(b_k) \leftarrow i_{b_k}, \quad \mathrm{SJ}(b_k) \leftarrow j_{b_k}, \quad (b_k = 1, 2, \cdots, N_b)$$

where values of i_{b_k} and j_{b_k} are dependent on the lattice structure under consideration. These arrays are always looked on as a reference table. As an example, a numbering of nearest neighbour sites and bonds between them on the 3×3 square lattice with the periodic boundary is shown in Fig. 1 and Table 1.

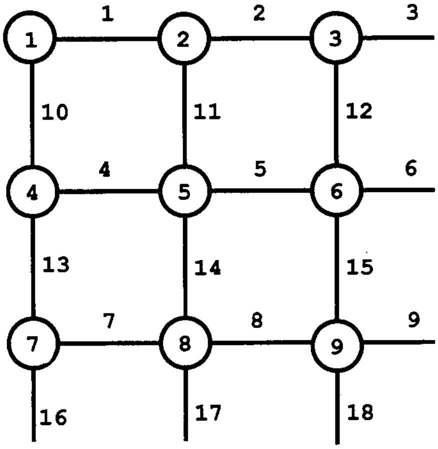

Fig 1. An example of the numbering of bonds and sites on 3×3 square lattice.

Table I. A look-up table for numbering bond b_k and it's nearest neighbour spin sites i_{b_k} and j_{b_k} on 3×3 square lattice.

b_k	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
SI	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9
SJ	2	3	1	5	6	4	8	9	7	4	5	6	7	8	9	1	2	3

3.2.1. *Algorithms*

Now we decompose $P(C_r)$ which is specified by r integers $b_1, b_2, \cdots, b_r$ into the produce of independent cycles as on the right hand side of Eq. (6). However, selecting a $P(C_r)$ and decomposing it into independent cycles can be made simultaneously in the computer program. In this respect it should be noticed that an operator $(i_{b_k} j_{b_k})$ acts to interchange spin states at sites i_{b_k} and j_{b_k}, and $P(C_r)$ is a permutation of order N. The simple algorithm for the cycle decomposition is derived from this fact. First, we initialize an array PCR of size N to $\mathrm{PCR}(i) \leftarrow i$ $(i=1, \cdots, N)$. Then we repeat r times the following transpositions

$$\mathrm{TMP} \leftarrow \mathrm{PCR}(i_{b_k}), \quad \mathrm{PCR}(i_{b_k}) \leftarrow \mathrm{PCR}(j_{b_k}), \quad \mathrm{PCR}(i_{b_k}) \leftarrow \mathrm{TMP} \ (k=1, \cdots, r)$$

where TMP is a temporary variable, and i_{b_k} and j_{b_k} which are the numbers stored in $\mathrm{SI}(b_k)$ and $\mathrm{SJ}(b_k)$, respectively, represent the nearest neighbour sites corresponding to a randomly selected bond number b_k. After repeating the procedure r times, we obtain the permutation $P(C_r)$ in the form

$$P(C_r) = \begin{pmatrix} 1 & 2 & 3 & \cdots & N \\ k_1 & k_2 & k_3 & \cdots & k_N \end{pmatrix}, \tag{24}$$

where $k_i (i=1, \cdots, N)$ is stored on the array PCR with the subscript i, that is, $k_i = \mathrm{PCR}(i)$. This is a very favorable property, because the operator $P(C_r)$ can be decomposed into independent cycles by only r times interchange. As a result the operator $P(C_r)$ is scanned once for all.[10] An example is given in the next subsection.

3.2.2. *An Example*

As an example for a randomly selected sequence $C_r = (1, 9, 9, 8, 11, 16, 3, 14, 4, 2)$ with $r=10$ a decomposition of $P(C_r) = (12)\ (97)\ 97)\ 89)\ (25)\ (71)\ 31)\ (58)\ (45)\ (23)$ into independent cycles is illustrated in Table II in the case of $N=3\times 3$. The correspondence between the bond number and the site numbers is shown in Fig. 1. In the case (a) of Table II, the permutation $\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ 3 & 7 & 5 & 9 & 4 & 6 & 2 & 1 & 8 \end{pmatrix}$ is produced by operating the transpositions from left to right. In the case (b), the permutation $\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ 8 & 7 & 1 & 5 & 3 & 6 & 2 & 9 & 4 \end{pmatrix}$ is produced by operating the transpositions from right to left. In order to obtain the traces only the number of independent cycles and their lengths are required. The direction of the operation of the transpositions is irrelevent. The cyclically decomposed form of $P(C_r)$ in case (a) is given by $(1\ 3\ 5\ 4\ 9\ 8)(2\ 7)(6)$ and in the case (b) by $(1\ 8\ 9\ 4\ 5\ 3)(2\ 7)(6)$ respectively. In the case (a) the decomposition is carried out as follows. First we see the most left integer

Table II. In case of N=3w×3, $r=10$, a decomposing $P(C_r)=(12)$ (97) (97) (89) (25) (71) (31) (58) (45) (23) for a randomly selected sequence C=(1, 9, 9, 8, 11, 16, 3, 14, 4, 2) into independent cycles is irustlated. Case (a) is given by scanning $P(C_r)$ from left to right. In the other hand case (b) does in the vice versa.

Case (a)

step	transposition	PCR
0-th	(1 2)	1 2 3 4 5 6 7 8 9
1-st	(9 7)	2 1 3 4 5 6 7 8 9
2-nd	(9 7)	2 1 3 9 5 6 9 8 7
3-rd	(8 9)	2 1 3 4 5 6 7 8 9
4-th	(2 5)	2 1 3 4 5 6 7 9 8
5-th	(7 1)	2 5 3 4 1 6 7 9 8
6-th	(3 1)	7 5 3 4 1 6 2 9 8
7-th	(5 8)	3 5 7 4 1 6 2 9 8
8-th	(4 5)	3 5 7 4 9 6 1 2 8
9-th	(2 3)	3 5 7 9 4 6 2 1 8
10-th		3 7 5 9 4 6 2 1 8

$$P(C_r)=\begin{pmatrix}1&2&3&4&5&6&7&8&9\\3&7&5&9&4&6&2&1&8\end{pmatrix}=(1\ 3\ 5\ 4\ 9\ 8)\ (2\ 7)\ (6)$$

Case (b)

step	transposition	PCR
0-th	(2 3)	1 2 3 4 5 6 7 8 9
1-st	(4 5)	1 3 2 4 5 6 7 8 9
2-nd	(5 8)	1 3 2 5 4 6 9 8 7
3-rd	(3 1)	1 3 2 5 8 6 7 4 9
4-th	(7 1)	2 3 1 5 8 6 7 4 9
5-th	(2 5)	7 3 1 5 3 6 2 4 9
6-th	(8 9)	7 8 1 5 3 6 2 4 9
7-th	(9 7)	7 8 1 5 3 6 2 9 4
8-th	(9 7)	7 8 1 5 3 6 4 9 2
9-th	(1 2)	7 8 1 5 3 6 2 9 4
10-th		8 7 1 5 3 6 2 9 4

$$P(C_r)=\begin{pmatrix}1&2&3&4&5&6&7&8&9\\8&7&1&5&3&6&2&9&4\end{pmatrix}=(1\ 8\ 9\ 4\ 5\ 3)\ (2\ 7)\ (6)$$

1 in the upper line and then the integer 3 right below in the lower line. We then look for the same integer 3 in the upper line and see the integer 5 right below in the lower line. Again, we look for the same integer 5 in the upper line and see the in-

teger 4 right below in the lower line. We repeat the same procedure until we encounter the integer 1 in the lower line. At this point we obtain a 6-cycle (1 3 5 4 9 8). Next, we look for the most left integer in the upper, except for the members of this cycle, and find the integer 2. We start from this integer, repeat the above-mentioned procedure and obtain a 2-cycle (27). Finally we obtain a 1-cycle (6).

Now we describe the detailed algorithm for this transformation. The permutation $\begin{pmatrix} 1 & 2 & 3 \cdots & N \\ k_1 & k_2 & k_3 \cdots & k_N \end{pmatrix}$ can be identified as follows. The integer i in the upper line of the permutation corresponds to the subscript i of the array PCR, and the integer k_i in the lower line corresponds to the contents of the array, PCR(i). Using the information of the array we carry out the same procedure as the above-mentioned example. It's algorithm is shown as follows, where the length of the j-th cycle is stored in CYCL-(J), the number of the cycles in NCYCL.

```
Step 1.  [Initialize.]  Set I ← 1.  Set J ← 1.
Step 2.  START ← I,  LEN ← 1,  IDX ← I  and  EL ← PCR(IDX).
Step 3.  [Is START equal to EL?]
    If yes, go to step 4.
    Otherwise, PCR(IDX) ← 0.  IDX ← EL.  LEN ← LEN+1.
    EL ← PCR(IDX).
    Repeat step 3.
Step 4.  PCR(IDX) ← 0.  CYCL(J) ← LEN.  J ← J+1.  I ← I+1.
Step 5.  [Is I greater than N?]
    If yes, go to step 6.
    [Is PCR(I) equal to 0?]
    If no, go to step 2.  Otherwise, I ← I+1.
    Repeat step 5.
Step 6.  NCYCL ← J—1.
```

This procedure, however, is not easy to be vectorized on the computer. It's improvement remains to be studied.

3.2.3 *How to evaluate Traces*

By the procedures given in the preceding subsections, for a given sequence C_r we can obtain the number of independent cycles (including a trivial one), $k(C_r)$, and the lengths of their cycles, l's.

As seen in Eqs. (9) and (23a), in order to obtain $\bar{A}_r$, only the total number of cycles, $k(C_r)$, for every randomly sampled C_r is required. To obtain $\bar{D}_r$ the sums of l^2 are necessary as seen in Eq. (16). For that purpose, executing the following preprocedure is convenien. We use an array KCR to sum up the frequency of $k(C_r)$. Equating a subscript i of the arrays to $k(C_r)$, a total number of independent cycles of a permutation $P(C_r)$, the contents of KCR(i) give the frequency of $P(C_r)$'s which have $i(=k(C_r))$ independent cycles among N_s samples. On the other hand, we use another array SL2 with the same subscript i to sum up the value of $\sum_{l=1}^{N} \nu_l l^2$ which

Table III. The numerical examples of $i(\equiv k(C_r))$, KCR(i) and SL2(i) with $N_s=10^5$ samples in the case of $r=5$, 150 and 10,000, respecively, for $N=10\times 10$ system. For details see the text.

r	i	KCR(i)	SL2(i)
5	95	94968	$1.050649\cdot 10^{7}$
	97	4995	$5.303880\cdot 10^{5}$
	99	37	$3.774000\cdot 10^{3}$
150	6	1	$8.304000\cdot 10^{3}$
	8	9	$5.528600\cdot 10^{4}$
	10	68	$3.531260\cdot 10^{5}$
	12	382	$1.725730\cdot 10^{6}$
	14	1280	$4.991066\cdot 10^{6}$
	16	3396	$1.163996\cdot 10^{7}$
	18	7053	$2.089823\cdot 10^{7}$
	20	11560	$2.951696\cdot 10^{7}$
	22	15451	$3.405946\cdot 10^{7}$
	24	17151	$3.296241\cdot 10^{7}$
	26	15647	$2.597454\cdot 10^{7}$
	28	12035	$1.738982\cdot 10^{7}$
	30	7926	$1.007998\cdot 10^{7}$
	32	4415	$4.931330\cdot 10^{6}$
	34	2194	$2.163426\cdot 10^{6}$
	36	938	$8.213300\cdot 10^{5}$
	38	340	$2.728040\cdot 10^{5}$
	40	117	$7.843800\cdot 10^{4}$
	42	26	$1.544600\cdot 10^{4}$
	44	7	$3.458000\cdot 10^{3}$
	46	4	$2.206000\cdot 10^{3}$
10000	2	10227	$8.281278\cdot 10^{7}$
	4	38555	$2.196572\cdot 10^{8}$
	6	35487	$1.521444\cdot 10^{8}$
	8	12991	$4.330956\cdot 10^{7}$
	10	2451	$6.585172\cdot 10^{6}$
	12	273	$6.034400\cdot 10^{5}$
	14	15	$2.681600\cdot 10^{5}$
	16	1	$1.272000\cdot 10^{4}$

is assigned to NU2. That is, SL2(i) give the sum of NU2 for $P(C_r)$'s which have $i=k(C_r)$ independent cycles among N_s samples. We note that for any two operators, $P(C_r)$ and $P(C'_r)$, NU2 for both operators do not always take the same value even though $k(C_r)=k(C'_r)$. In the computer program given by Listing 1, the following is done with use of both arrays.

$$\mathrm{KCR}(i) \leftarrow \mathrm{KCR}(i)+1 \quad \text{and} \quad \mathrm{SL2}(i) \leftarrow \mathrm{SL2}(i)+\mathrm{NU2}, \quad (i=1, \cdots, N),$$

where the array KCR is declared as type INTEGER and array SL2 and variable NU2 as type REAL in the listing. Numerical examples are shown in Table III where for $N=10\times 10$ $i(\equiv k(C_r))$, KCR(i) and SL2(i) for r=5,150 and 10,000, respectively, are given for $N_s=10^5$. For example, for $i=24$ in the case of $r=150$, KCR(i)$=17151$ gives the frequency of $P(C_r)$'s having $k(C_r)=24$. SL2(i)$=3.296241\cdot 10^7$ is the sum of NU2, $\sum_{l=1}^{N} \nu_l l^2$ in Eq.(16), having KCR(i)$=17151$ among $N_s=10^5$. On obtaining $\bar{A}_r$ we have to evaluate $k(C_r)$-th power of two in Eq.(9) for all of N_s samples. Using the above-mentioned frequency we can obtain $\bar{A}_r$ by evaluatting only the power of two for distinct $k(C_r)$'s. That is,

$$\bar{A}_r = \frac{1}{N_s} \sum_i \mathrm{KCR}(i)\cdot 2^i ,$$

In the numerical example of $r=150$ in Table III, $\bar{A}_r$ is obtained by evaluating only 21 $A(C_r)$'s. On the other hand, we can obtain $\bar{D}_r$ by multiplying these twenty one $A(C_r)$'s by the corresponding values of SL2, respectively. That is,

$$\bar{D}_r = \frac{1}{N_s} \sum_i \mathrm{KCR}(i)\cdot 2^i \cdot \mathrm{SL2}(i).$$

4. The Interpolation Method

In the previous section, we have given a full detail of the method for evaluating the expansion coefficients. It is however time consuming to evaluate even first some hundreds of terms for a large N by this method. So, as the next step, the rest of the coefficients of r less than r_M are evaluated by the interpolation with use of $\bar{A}_{r_i}$ and $\bar{D}_{r_i}$ $(i=1, 2, \cdots, M)$. Since $log A_r/N(\equiv y_r)$ is founds to be a monotonically decreasing function of $r/N=(\equiv x_r)$, we assume, a rational function, that is,

$$y_r = \frac{\beta_0+\beta_1 x_r+\beta_2 x_r^2+\cdots+\beta_m x_r^m}{\alpha_0+\alpha_1 x_r+\alpha_2 x_r^2+\cdots+\alpha_n x_r^n}, \tag{25}$$

to interpolate $\bar{A}_r$ for any r less than or equal to r_M, where $\alpha_0=1$ and $m+n+1=M$. M

constants $\alpha_1, \cdots, \beta_m$ are determined by numerically solving linear simultaneous equations

$$\sum_{i=0}^{m} \beta_i x_{r_k}^i - y_{r_k} \sum_{i=0}^{n} \alpha_i x_{r_k}^i = 0, \qquad k=1, 2, \cdots, M. \tag{26}$$

Anyway, this interpolation formula is found to be very powerful, because *only less than ten* coefficients suffice to obtain some hundreds of thousands of interpolated expansion coefficients. This reduces computing time tremendously. But since a great many coefficients are needed to evaluate precise thermodynamic quantities, a mass of the magnetic disk area is needed to save the coefficients. In order to save for the used disk area, we should save at least an UNFORMATTED output-file. Futhermore, using interpolated coefficients up to very high terms for the larger system size, thermodynamic quantities can be evaluated rather precisely down to comparatively low temperatures. Thermodynamic quantities are obtained by using Eqs. (17) to (22). In order to obtain precise quantities, the lower the τ, the more terms are needed. That is, r becomes larger. As seen in Eq.(10), for small τ and large r, the values of $a_r\tau^{-r}/r!$'s are so large that they cannot be repesented by the existing floating-point data types. In order to overcome this difficulty, we introduced an implementation of the following type of data and used it for obtaining the results in II. That is, to handle a very large number as a floating-point number, a variable, say AE, of type REAL is used for a mantissa and a variable, say AE, of type INTEGER for an exponent. We define that the decimal point is at the left-hand end of the mantissa and the leading digit in its mantissa is always non-zero, except for number zero. On converting to this form by using common logarithms, the excess of the exponent has to be added to AE for the exponent part. For example, a number $295.1696 \cdot 10^{17151}$ becomes AM←0.2951696, AE←17154. In accord with this form, numbers with the exponent in the range of −2,147,483,648 to +2,147,483,6487 as decimal floating-point numbers can be handled on any machine. A full computer program (including these arithmetic subroutines) will be given elsewhere.[11] A similar implementain was recently published independently by O.Portilho.[12]

5. Comments on the method

In this section, some technical problems on using this method are described. Let the r_{Max}-th term be the term giving the maximum value among $a_r\tau^{-r}/r!$ (see Eq. (10)) for given τ. We find empirically that r_{Max} is roughly by $N_b/2\tau$ for τ greater than 1.0 and N_b/τ for τ less than 1.0. From this fact, in order to obtain accurate values of thermodynamic quantities of the finite system at temperatures higher than τ^*, r_M has to be chosen greater than N_b/τ^*. Another check on r_M is provided by examining how close $\bar{A}_{r_M}$ is to $N+1(\equiv A_\infty)$ (see Appendix in I.). Fortunately, the $\bar{A}_{r_M}$ whose value is close to $N+1$ can be evaluated with comparatively small sampling times. On the other hand A_1 is exactly given by 2^{N-1}. Therefore two integers r_1, r_2, $\cdots$, r_M of the interpolating expansion coefficients $A_{r_1}, A_{r_2}, \cdots, A_{r_M}$, that is r_1 and r_M, are chosen in this way. The integers r's of the rest of the coefficients have to be chosen carefully so that the A_r's obtained by the interpolation decrease monotonous as r increases. We empirically found that if one uses more than ten coefficients as interpolating data, y_r in Eq. (25) has a tendency to have a pole. Therefore, in order to avoid this, it is recommended to select two sets of these coefficients, i.e., one for τ higher than 1.0 and another for sufficiently low temperatures (also see Section 3 in II). Next, concerning statistical accuracy of the MC values $\bar{A}_r$ and $\bar{D}_r$ we have to check the convergence of these MC values as a function of sampling times N_s as shown in Fig. 1 in I and Figs. 1, 2 and Table II in II. Especially, for $r \sim N$ and the large N this check is very important since large sampling times are required owing to a statistically large fluctuation arising from the fact that the $P(C_r)$ consists of cycles of various lengths, short to long.

6. Discussion

In the previous works[1], we showed unexpectedly precise numerical results for linear ring Heisenberg ferromagnets of 10, 20, 30 and 128 spins and the square lattice Heiseberg ferromagnet of 10×10, 20×20 and 30×30 spins making use of the detailed procedure described in this paper. Especially, in regard to the values of the susceptibility on linear systems of 128 spins it was shown in Table VIII in II that the agreement between our results and Baker et al.'s[13] at high temperature and Takahashi's[15] at low temperature were excellent. Also as seen from Table VII in II, for the susceptibility for temperatures $\tau \geq 2.0$, ours of the 10×10 spin system nicely agreed with Baker et al.'s[14]. In a recent paper[16] Takahashi showed that the agreement between his and our values of the thermodynamic quantities was rather good for the 10×10 square lattice system at low temperature. However, for 20×20 and 30×30 spin systems, the agreement between both was not so good. This seems partly due to the fact that the finiteness of the system was important for 2D ferromagnet.[16] However, it is true that the present simple approach with a crude Monte Carlo method provides unexpectedly precise results on thermodynamic quantities of finite systems of spin 1/2 isotropic Heisenberg ferromagnets for rather low temperatures.

The present method so far being described consists of two parts. One is to evaluate the expansion coefficient a_r for several r's, and the other is to obtain the rest of the expansion coefficients with use of the rational polynomial interpolation. Concerning the former, the trace can be rather easily evaluated with use of the Dirac exchange operator in case of the Heisenberg system of spin 1/2. A similar procedure can also be applied to model systems described by the Schrodinger exchange operator of the arbitrary spin S. For a system which has no permutation property, it is necessary to develop a method for computing the trace. Application of our method to XY-model is now being undertaken.[17]

Listing. 1

```
C     ********************
        PROGRAM FR2D
C     ********************

      PARAMETER (MS1=1024, MS2=2*MS1)
      INTEGER BK, N, NB, NBMN, NCYCL, NN, NM1, NS
      INTEGER I, IDX, II, I1, J, J1
      INTEGER EL, R, TMP
      INTEGER CYCL (MS1), PCR(MS1), KCR (MS1), SI (MS2), SJ (MS2)
      REAL NU2, TL2, SL2 (MS1)

    Variables and constants for Portable Random Number Generators RAN2

       INTEGER IY, IDUM, IR (97)
       PARAMETER (M=714025, IA=1366, IC=150889, PM=1.4005112E-6)

    The previous saved seeds in the file FR2D. RAN are restored kinto the shuffling
    array IR.

       OPEN (UNIT=3, FILE=' FR2D. RAN', STATUS=' OLD')
       READ (3,*) IY, IDUM
       READ (3,*) (IR(I), I=1, 97)
       CLOSE (3)

       WRITE(*,*)' Input linear spin size N (N×N):'
       READ (*,*) N
       NN=N*N
       NB=2*NN
```

```
      NM1=N-1
      NBMN=NB-N
```

The look-up table of the nearest neighbour spin sites corresponding to bond-number b_k for $N \times N$ square lattice with the periodic boundary condition is created.

```
      DO 110 BK=1, NB
        IF (BK. LE. NN) THEN
          IF (MOD (BK, N). EQ. O) THEN
            SI(BK)=BK
            SJ(BK)=BK-NM1
          ELSE
            SI(BK)=BK
            SJ(BK)=BK+1
          ENDIF
        ELSE
          IF (BK. GT. NBMN) THEN
            SI(BK)=BK-NN
            SJ(BK)=BK-NBMN
          ELSE
            SI(BK)=BK-NN
            SJ(BK)=SI(BK)+N
          ENDIF
        ENDIF
110   CONTINUE
200   WRITE(*,*) 'Input a term and # of samples.'
      READ (*,*) R, NS
```

For every $k(C_r)$ of Eq. (9) and the sum of l^2 of Eq. (16) to obtain the MC values $\bar{A}_r$ and $\bar{D}_r$ per N_s samples are produced.

```
      DO 210 I=1, NS
```

The array PCR for the permutation operator $P(C_r)$ is initialized.

```
        DO 220 I1=1, NN
          PCR(I1)=I1
220     CONTINUE
```

In the following DO loop, an operator $P(C_r)$, is produced and simultaneously in decomposed in the form of the left hand side of Eq. (6) in the text.

```
      DO 230 J=1, R
```

The Portable Random Number Generator RAM2 by Press et al. is used as an inline code. See the following textbook for the particulars.
W.H. Press, B.P. Flannery, S.A. Teukolsky and W.T. Vetterling, *Numerical Recipes* (Cambridge University Press, Cambridge, 1987) Chapter 7, page 197.

```
          L=1+(97*IY)/M
          IF(L.GT.97.OR.L.LT.1)PAUSE
          IY=IR(L)
          IDUM=MOD(IA*IDUM+IC, M)
          IR(L)=IDUM
          BK=1+INT(NB*(IY*RM))
          TMP=PCR(SI(BK))
          PCR(SI(BK))=PCR(SJ(BK))
          PCR(SJ(BK))=TMP
230   CONTINUE
```

The operator $P(C_r)$ is decomposed into independent cycles in the right hand side of Eq. (6) and $k(C_r)$ and the sum of l^2 for each C_r is obtained.

```
      I1=1
      J1=1
240   START=I1
      LEN=1
      IDX=I1
      EL=PCR(IDX)
242   IF (START.EQ.EL) GOTO 244
      PCR(IDX)=0
      IDX=EL
      LEN=LEN+1
      EL=PCR(IDX)
      GOTO 242
244   PCR(IDX)=0
      CYCL(J1)=LEN
```

```
      J1=J1+1
      I1=I1+1
246   IF (I1.GT.NN) GOTO 248
      IF (PCR(I1).NE.O) GOTO 240
      I1=I1+1
      GOTO 246
248   NCYCL=J1-1
      NU2=0.0D0
      DO 250 J1=1, NCYCL
        TL2=CYCL(J1)
        MU2=NU+TL2*TL2
250   CONTINUE
      KCR(NCYCL)=KCR(NCYCL)+1
      SL2(NCYCL)=SL2(NCYCL)+NU2
210   CONTINUE
```

The frequency based on grouping by $k(C_r)$'s for every $P(C_r)$ and the sum of each group by the sum of l^2 are produced.

```
      OPEN (UNIT=1, FILE='FR2D.DAT', ACCESS='APPEND', STATUS
      ='OLD')
      WRITE(*,*) NN,R,NS
      WRITE(1,*) NN,R,NS
      DO 260 I=1, NN
        IF (KCR(I).EQ.O) GOTO 260
        WRITE(*,'(2I10,E14.7)') I,KCR(I),SL2(I)
        WRITE(1,'(2I10,E14.7)') I,KCR(I),SL2(I)
260   CONTINUE
```

The data per N_s samples are delimited by a dummy $k(C_r)$ $(=-1)$.

```
      WRITE(1,'(2I10,E14.7)')-1,0,0.0
      CLOSE(1)
```

The contents of the array IR are saved for the file FR2D.RAN for the next time.

```
      OPEN (UNIT=3,FILE='FR2D.RAN',STATUS='NEW')
      WRITE(3,*) IY,IDUM
      WRITE(3,*) (IR(I), I=1,97)
```

```
CLOSE (3)

END
```

References

1) S. Kadowaki and A. Ueda: Prog. Theor. Phys **75**, 451 (1986); ibid **78**, 224 (1987). (referred to as I) and ibid **82**, 493 (1989) (referred to as II).

2) W. Opechowski: Physica **4**, 181 (1937).
G.S. Rushbrooke and P. J. Wood: Mol. Phys. **1**, 257 (1958).
C. Domb and M. F. Sykes: Phys. Rev. **128**, 168 (1962).
G.S. Rushbrooke, G.A. Baker, Jr. and P.J. Wood: *Phase Transition and Critical Phenomena ed.* Domb and Green (Academic Press, London, 1974), p.245 C. Domb: Advan. Phys. **9** Nos. 34 and 35 (1960).
See references cited in these papers and textbooks for details of the applications using HTEM.

3) D.C. Handscomb: Proc. Camb. Phil. Soc. **58**, 594 (1962); *ibid* **40**, 115 (1964).

4) J.W. Lyklema: Phys. Rev. Lett. **49**, 88 (1982); Phys. Rev. 27, 3108 (1983).
J.W. Lyklema: *Monte Carlo Method in Quantum Problem*, ed. M.H. Kalos (Reidel, Boston, 1982), p148.

5) S. Chakravarty and D.B. Stein: Phys. Lett. **49**, 582 (1982).

6) D.H. Lee, J.D. Joannopolos and J.W. Negele: Phys. Rev. **B30**, 1599 (1984).

7) E. Manousakis and R. Salvador: Phys. Rev. Lett. **60** 840 (1988).

8) P.A.M. Dirac: *The Principles of Quantum Mechanics* 4th ed. (Clarendon, Oxford, 1958), p. 222.

9) M. Hammermesh: *Group Thery and Its Application to Physical Problem* (Addison-Wesley, 1964).
J. Riordon: *An Introduction to Combinatorial Analysis* (John Wiley, Canada, 1958), p.66.

10) D.E. Knuth: *The Art of Computer Programming*, Vol. 1
(Addison-Wesley, Reading, Massachusetts, 1972), p.169.

11) S. Kadowaki and A. Ueda: (to be published.)

12) O. Portilho: Comp. Phys. Communi. **59** 345 (1990).

13) G.A. Baker, G.S. Rushbrooke and H.E. Gilbert: Phys. Rev. **A135**, 1272 (1964).

14) G.A. Baker, H.E. Gilbert, J. Eve and G.S. Rushbrooke: Phys. Lett. **14**, 207 (1967).

15) M. Takahashi: Prog. Theor. Phys. Suppl. No.87, 233 (1986).

16) M. Takahashi: Prog. Theor. Phys. **83** No.5 (1990), 815.

17) K. Katsuda, S. Kadowaki and A. Ueda: (in preparation.)