# Monte Carlo Technique for Very Large Ising Models

C. Kalle<sup>1</sup> and V. Winkelmann<sup>1</sup>

Received February 18, 1982

Rebbi's multispin coding technique is improved and applied to the kinetic Ising model with size 600 \* 600 \* 600. We give the central part of our computer program (for a CDC Cyber 76), which will be helpful also in a simulation of smaller systems, and describe the other tricks necessary to go to large lattices. The magnetization M at  $T = 1.4 * T_c$  is found to decay asymptotically as  $\exp(-t/2.90)$  if t is measured in Monte Carlo steps per spin, and M(t = 0) = 1 initially.

**KEY WORDS:** Monte Carlo; Glauber kinetic Ising model; Multispin coding; CDC computers.

# 1. INTRODUCTION

The Ising magnet on a simple cubic lattice serves as a model for many cooperative phenomena. Its simulation on a computer uses the Metropolis Monte Carlo method,<sup>(1)</sup> usually with Glauber kinetics.<sup>(1)</sup> Rebbi and co-workers<sup>(2)</sup> improved the efficiency of this computer simulation by "multispin coding"; a complete program was published by Zorn *et al.*,<sup>(2)</sup> and was further improved by Ottavi and Chakrabarti *et al.*<sup>(4)</sup> These latter authors used it also for a simulation of large lattices up to 360 \* 360 \* 360 \* 360. The present work started from the program used in that paper,<sup>(4)</sup> reduced its computer time by about 40%, and applied it to larger systems. Basically we made a feasibility study only; the application to unsolved problems is left for the future.

In CDC 6000 and 7000 series computers each memory unit ("word") has 60 bits; in a (spin-1/2) Ising model, one bit is sufficient to represent a single spin. Multispin coding allows three bits for every spin and thus

<sup>&</sup>lt;sup>1</sup> University of Cologne, Computing Center, Robert-Koch-Strasse 10, D-5000 Cologne 41, West Germany.

makes it possible to count the number of antiparallel neighbor spins, from zero to six in the simple cubic lattice, with these three bits. Thus 20 spins (or 20 interaction energies) can be stored in one word and calculations for these can be made with one word-oriented instruction. Therefore, compared with conventional one-word-per-spin methods,<sup>(1)</sup> the memory required is reduced by a factor of 20, the computer time by a factor of 2 (Zorn *et al.*<sup>(2)</sup>). Basically the program performs an exclusive-or operation (XOR) between a word representing 20 spins and their neighbors; the result gives in each of the 20 three-bit parcels the number of antiparallel neighbors,<sup>(2)</sup> and thus the interaction energy. Thereafter each spin has to be flipped separately according to the probability with a random number (FORTRAN function RANF) distributed homogeneously between zero and one. (For computers with 32-bit words, the advantages of multispin coding are less drastic.)

In the following section we describe the tricks to speed up computation in our CDC Cyber 76 M and include a list of that part of the program. Section 3 describes how a hierarchy of three storage devices is used for the simulation of systems larger than 100 \* 100 \* 100. Section 4 gives and discusses our results.

# 2. THE CENTRAL ALGORITHM

For each Monte Carlo step per spin, the computer program examines regularly every spin with coordinates i, j, k and flips and checks if it should be flipped (i, j, k = 1, 2, ..., L). The outermost loop with variable K is left out in our program listing for simplicity (see Section 3), the second loop (DO 1 J = 1, L) starts our program listing in Table I. As the reader may see, periodic boundary conditions were employed. The innermost loop is reduced to a loop over LL = L/20 steps only, since in multispin coding we will treat with II = 1 the 20 spins I = 1, 1 + LL, 1 + 2\*LL, ..., 1 + 119\*LL, with II = 2 the 20 spins  $I = 2, 2 + LL, 2 + 2*LL, \ldots, 2 + 19*LL$ , until finally with II = LL we treat the 20 spins I = LL, 2\*LL,  $3^*LL, \ldots, 20^*LL (= L)$ . The two statements after labels 10 and 11 incorporate the special cases II = 1 and II = LL, where the left or right neighbors in the lattice row are not characterized by  $II \pm 1$  but have to be formed by shift operations.<sup>(3)</sup> Essentially we first calculate, apart from differences due to periodic boundary conditions, the sum of the six neighbors and the central spin:

$$IEN = IS(II - 1, J, K) + IS(II + 1, J, K) + IS(II, J - 1, K)$$
  
+ IS(II, J + 1, K) + IS(II, J, K - 1) + IS(II, J, K + 1)

For reasons of programming technique we denote K + 1 and K - 1 by the

С

1

CONTINUE

_	
	DO 1 J = 1, L JP1 = J + 1  IF(JP1.GT.L) JP1 = 1
	JM1 = J - 1 \$ IF(JM1.EQ.0) $JM1 = L$
	DO 1 II = 1, LL
	IF(II.EQ.1) GOTO 10
	IF(II.EQ.LL) GOTO 11
	IEN = ISC(II + 1, J) + ISC(II - 1, J)
	GOTO 12
10	CONTINUE
	IEN = ISC(2, J) + SHIFT(ISC(LL, J), 57)
	GOTO 12
11	CONTINUE
	IEN = ISC(LLM1, J) + SHIFT(ISC(1, J), 3)
12	ICI = ISC(II, J)
	IEN = IEN + ISC(II, JM1) + ISC(II, JP1) +
	• IS(II, J, PLLCMU) + IS(II, J, PLLCML) + ICI
	IEN = XOR(ICI.OR.SHIFT(ICI, 1).OR.SHIFT(ICI, 2), IEN) + IEN1
С	IEN COUNTS FOR THE 20 SPINS IN ICI HOW MANY OF THEIR
С	NEIGHBORS ARE ANTIPARALLEL TO THEM (EXCLUSIVE OR)
С	ICH MARKS THOSE SPINS WHICH HAVE TO BE FLIPPED
	ICH = 0
	DO 2 J1 = 1, 10
	IF(RANF(J1).LE.EX(IEN.AND.7)) ICH = ICH.OR.7
	ICH = SHIFT(ICH, 3)
	IEN = SHIFT(IEN, 3) IF(RANF(J1).LE.EX(IEN.AND.7)) ICH = ICH.OR.7
	IF(RANF(J1):LE:EX(IEN:AND.7)) ICH = ICH.OR.7 ICH = SHIFT(ICH, 3)
2	IEN = SHIFT(IEN, 3)
Ĉ	SHIFT PRODUCES A CIRCULAR SHIFT TO THE LEFT BY 3 BITS
~	ICI = ISC(II, J) = XOR(ICI, ICH.AND.IEN1)
	M = M + COUNT(ICI)

Table I. Central Part of Computer Program, Showing for One Sweep through the Lattice the Loops in Two of Three Dimensions in the L \* L \* L Lattice<sup>a</sup>

<sup>a</sup>RANF is a random number generator and EX(n) the probability to flip a spin if n - 1 of its neighbors are antiparallel. Basically, PLLCMU = K - 1 and PLLCML = K + 1 if the investigated plane K is stored in ISC(II, J). This part can be applied to smaller systems, at least L = 60. IEN1 adds unity to every 3-bit parcel.

COUNT COUNTS THE NUMBER OF SET BITS IN A COMPUTER WORD

integer "plane-in-large-core-memory (LCM)" variables PLLCML and PLLCMU (see Section III), and we store central plane K also in small-corememory, the usual working storage, as a copy of the plane K resident in LCM, such as we can reference IS(II, J, K) by ISC(II, J). ICI is an abbreviation<sup>(3)</sup> for IS(II, J, K).

The crucial statement

# IEN = XOR(ICI.OR.SHIFT(ICI, 1).OR.SHIFT(ICI, 2), IEN) + IEN1

gives in every 3-bit group the number of antiparallel neighbors of the central spin in ICI, as the reader may check in an example. [The two "OR"-operations give 111 for every up spin (001) and 000 for every down spin (000).] This method works if each spin interacts with six neighbors and has to be modified for different numbers of interacting neighbors. The addition of IEN1 to this result adds unity to every number of antiparallel neighbors; this IEN1 has 001 in each of its 20 3-bit parcels. Earlier work<sup>(2-4)</sup> used an XOR operation for each of the six directions of interaction, which is less effective.

The statements "IF (RANF(J1).LE.EX(IEN.AND.7)) ICH = ICH. OR.7" will result in ORing a 7 (= 111) to the changer word ICH if the spin has to be flipped, and a 000 otherwise. Since the number of hardware registers in the computer is exhausted by this loop construction, it was not useful to mark spins to be flipped by an additional constant 1 (= 001). which would be more convenient for the flip process. In contrast to earlier work [2-4], we investigated two spins during one trip of loop labeled 2, since the small number of instructions allows the loop still to reside in the Cyber 76 instruction stack. In order to get a feeling for the time consumed in this loop, the reader may consider that each machine instruction in this innermost loop will be executed 216.000.000 times for each Monte Carlo step per spin (600 \* 600 \* 600 system). The cycle time of a Cyber 76 computer is 27.5 nsec, thus to save one cycle's time in this loop means saving 5.8 sec per Monte Carlo step. So we took great care to optimize the code of this loop. Thereafter the statement ICI = XOR (ICI, ICH. AND.IEN1) produces the changed spin word; and finally M = M + MCOUNT(ICI) counts the number of spins 001 in the changed configuration and so gives the magnetization more simply than in earlier versions.

We found that in our test runs for 240\*\*3 the CPU time for one Monte Carlo step was reduced to 14 sec compared with 22 sec in the program of Chakrabarti *et al.*<sup>(4)</sup> given to us. For the 600\*\*3 system we needed less than 230 sec for one sweep through the lattice. Another factor of 3 can be gained in speed if one takes  $\exp(-4J/k_BT) = 1/2$  corresponding to  $T/T_c = 1.279$  and treats 20 spin flips simultaneously<sup>(5)</sup>.

# 3. HIERARCHY OF MEMORIES FOR VERY LARGE SYSTEMS

Chakrabarti *et al.*<sup>(4)</sup> used already disk storage to store larger systems up to 360\*\*3, with test runs up to 600\*\*3. Information to and from the disk

#### Monte Carlo Technique for Very Large Ising Models

storage can be transferred asynchronously by the FORTRAN "BUFFER IN" and "BUFFER OUT" statements; using this method a program may perform computation and input/output at the same time. We corrected an error in the given program<sup>(4)</sup>, due to which parts of the boundary spins were always down, and further refined the method.

We used the three available types of memory in a strongly hierarchical way: SCM, the small-core memory; LCM, the large-core memory; and RMS, the disk storage system. As mentioned above, the plane K is stored with dimension ISC(LL, L) in SCM, this plane K, its neighbors K + 2, K + 1, K - 1 and the first plane K = 1 are stored in direct access LCM ("LEVEL 2") in the area IS(LL, L, 5). Information between SCM and LCM was transferred by the fastest possible method, the block copy (FORTRAN "MOVLEV" routine). The third index n of IS(II, J, n) was denoted as n = PLSCM for the central plane K, as n = PLLCMU for the upper neighbor plane K - 1, n = PLLCML for the lower neighbor plane K + 1, and n = PLLCMI for the input plane K + 2 read in from disk storage.

At this point, it may be interesting to the reader, in order to understand the following, to learn a little about single-word access to LCM in the Cyber 76 computer.

If a program references a word in LCM this and the following 15 words are read from core into a hardware register. Successive references to words following that word are satisfied from that register, which is quite fast. So only every 16 times a real storage access is required, if consecutive words are read using single-word access to LCM. This is the clue to make LCM access fast. Since the minimum LCM configuration consists of two banks of LCM, each having such a hardware register, this configuration is presently available at the Cologne Cyber 76, the only possibility to use LCM in the fastest way is to read the PLLCML plane using one bank and to read the PLLCMU plane using the other bank. Thus computation time is optimal if the product of LL = L/20 and L is an odd multiple of 16, allowing L = 200, 280, 360, 440, 520, 600, and 680 for the Cologne configuration. (L = 120 and smaller systems can be computed without RMS usage.)

All L planes are stored on RMS, accessed by "BUFFER IN" and "BUFFER OUT" statements. We did not test if we could fit 680\*\*3 into the disk storage system under normal multiuser conditions; for 600\*\*3 we consumed already nearly one fifth of the total available RMS space, that is about 20 million (60-bit) words. (One may save memory by a factor of 3, losing in speed by a factor of 1.6, if one compresses three 20-spin words into one before storing them on the disk. Only test runs were made, however.<sup>(5)</sup>) It is not sufficient to store only the planes K - 1 and K + 1 in LCM, because, as we mentioned above, we want to do asynchronous input/output, so during the time the lowest plane K + 2 is read in from RMS by "BUFFER IN," the plane K + 1 is already needed for computation, and during the time the uppermost plane K - 1 is output to RMS, its LCM storage can be used by spins of the upper plane. Table III explains this usage of plane indices. Effectively three processes are done at a time: input of a new plane, computation, and output of an already newly computed plane.

When we have investigated one plane, we move to the lower neighboring plane not by shifting the spins IS(II, J, n) within storage but merely by rotating the pointers (third index n), e.g., setting PLSCM = PLLCML. If these variables are set up as PLLCMU = 4, PLSCM = 2, PLLCML = 1, and PLLCMI = 3 the single-word access to LCM via the two banks is used best. Table II indicates schematically the way the information is shifted between memories. Additional statements are required for the special cases K = 1, K = L - 1, K = L, K = L + 1 and during the first Monte Carlo sweep through the lattice. After each such Monte Carlo step per spin, the magnetization  $(M - L^{**3}/2)/L^{**3}$  is printed along with other useful information.

To protect the computation against a breakdown of the computer, the (very limited) output was also printed on the dayfile (which is usually not lost), and the RMS information was saved for later reuse. Thus we could

	M = -L3/2
	DO 8 K = 1, L
	CALL MOVLEV(IS(1, 1, PLSCM), ISC(1, 1), LLL)
	•
	•
	Computation as in Table I
	•
	•
	CALL MOVLEV(ISC(1, 1), IS(1, 1, PLSCM), LLL)
С	MOVE SCM PLANE BACK TO LCM
	SCRATCH = PLLCMU
	PLSCM = PLLCML
	PLSAVEI = PLLCMI
	PLLCML = PLONE
	PLLCMI = SCRATCH
	BUFFEROUT (OTAPE, 1) (IS(1, 1, PLLCMU), IS(LL, L, PLLCMU))
	BUFFERIN (ITAPE, 1) (IS(1, 1, PLLCMI), IS(LL, L, PLLCMI))
8	CONTINUE

Table II.	Schematic Listing of Storage Handling Statements
-----------	--

Monte Carlo Technique for Very Large Ising Models

Disk storage	
	PLLCMU (output)
	PLSCM (used in SCM)
	PLLCML
	PLLCMI (input)
Disk storage	

Table III. Sketch of Plane Index Usage for Asynchronous Disk Input/Output during Computation

stretch the calculation over an extended period of time, and were not forced to compute all steps during one long run. No special arrangements with the general operating system were necessary, our program ran under usual conditions, as did numerous other programs at that time.

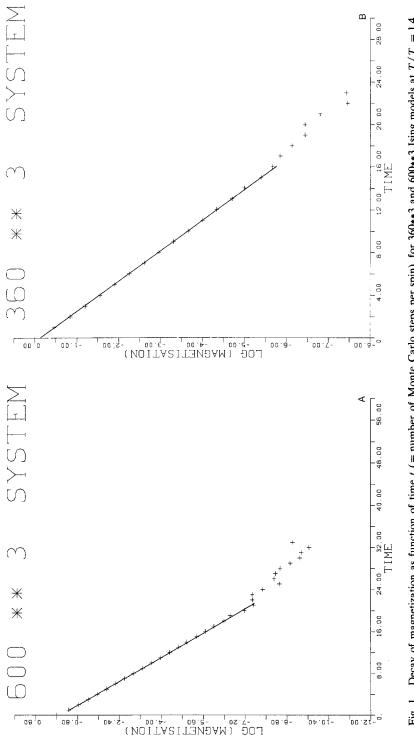
### 4. RESULTS

Figure 1 shows that the magnetization after t Monte Carlo steps per spin at a temperature 40% above the critical temperature. We see that the magnetization decays for large t as  $\exp(-t/\tau)$  with  $\tau = 2.90$ , until finally fluctuations take over. This decay is seen here much better than in Ref. 4, either due to an unfortunate fluctuation there or due to the abovementioned programming error in Ref. 4. For small times, deviations occur as expected. For the initial slope -dM/dt [at t = 0, M(t = 0) = 1] is equal to the probability of flipping one spin down when all neighbor spins are up. That probability remains finite at the critical temperature whereas  $1/\tau$ vanishes there<sup>(4)</sup>. [The flip probability was taken as

$$e^{-\beta\Delta E}/(1+e^{-\beta\Delta E})$$

where  $\beta = 1/k_B T$  and  $\Delta E$  is the energy change of the flip.]

Thus we have simulated the decay of magnetization in a very large Ising lattice on a general-purpose computer under normal operating conditions, using less than four minutes central processor time for every Monte Carlo step per spin in a 600\*600\*600 lattice. While we used many features special to large CDC computers, the whole program was written in FOR-TRAN. It remains to be seen what new results can be obtained from large systems; for this purpose we can send our complete program (200 statements) to the interested reader.





UKLH SCOPE 2,115 LVL 53342, 28/89/01 21,87,34, 94/02/82 7235 SY5 DEVICES 444/ 7/PF FL54377A FLL=2808K MX8=176K MXL=40PK MXD=18088 MMLHA 25 CPU SECOND OFTUIN 22,44513 00070,084 M72, -48002,57MF2, -48002,57MF2, -48002,57MF2, -48002,57MF2, -48002,57MF2, -48002,57MF2, -48002,57MF2, -48002,57MF2, -48002,57MF2, -28177, -281 CP SECONDS COMPLATION TIME 22,44513 00070,935 M72, -281 CP SECONDS COMPLATION TIME 22,4453 00080,955 M72, -281 CP SECONDS COMPLATION TIME 22,4451,00080,357 M72, -281 CP SECONDS EXECUTION FL, -281 CP SECONDS EXECUTION FL, -1800,57M C2, -484 00081,560 M72, -184 M10, -6 -484 -25/45/61 -2817,5, -6 -484 -27/45/61 -2817,5, -6 -484 -27/45/61 -27/25,460 -27/45, -184 0007,57M 72, -17/5, -5 -106 0007,57M 72, -17/5, -5 -160 -2007,57M 72, -184 0007,57M 72, -17/5, -5 -28,460 -480 -27/5, -27,460 -480 -27/5, -27,460 -480 -27/5, -27,460 -480 -27/5, -27,460 -480 -27/5, -27,460 -27,27 -27,27 -28 -27,460 -27,27 -2	APU02SU AF DC=40 II IIII II II II II II II II II IIIII II		I         IIIIII         IIIIIII         IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	
<pre>HM.HM.SS CPU SECOND ONIGIN</pre>	UKLN SCOPE 2.1.5 L	VL 533	A2 , 28/09/01 21.07.34 94/02/A2 A2035	
22.49.5%         DUT KUEN NG3/BE1.4 LEV. 506 0%, P2.1992           22.49.5%         DUBUN, UGA HT,         ACCOUNT, ARUR2, STM22,           22.49.5%         DUBUN, UGA HT,         ACCOUNT, ARUR2, STM22,           22.49.5%         DUBUN, UGA HT,         ACCOUNT, ARUR2, STM22,           22.49.5%         DUBUN, UGA HT,         DUBUN, UGA HT,           22.49.6%         DUBUN, UGA HT,         DUBUN, UGA HT,           22.49.6%         DUBUN, UT,         DUBUN, UGA HT,           22.49.6%         DUBUN, UT,         DUBUN, UGA HT,         DUBUN, UGA HT,           22.49.6%         DUBUN, UT,         DUBUN, UT,         DUBUN, UT,           22.49.6%         DUBUN, UT,         DUBUN, UT,         DUBUN, UT,           22.49.6% <thdubun, th="" ut,<="">         DUBUN, UT,         DUBUN,</thdubun,>	SYS DEVICES #44/ 7	/HF FL	Sa377k FLL=2000K MXS=170K MXL=400K MXB=1000B	
22,49,51 @000mi_000 H72, 22,49,51 @000mi_000 H72, 22,49,51 @000mi_020 W57, .221 CP SECONDS COMPILATION TIME 22,49,53 @000mi_057 W72, LD010 + LS REQUIPED TO LOAD - Reliad5 OU_COG 22,49,53 @000mi_057 W72, LD010 + LS REQUIPED TO LOAD - Reliad5 OU_COG 22,49,57 @000mi_050 W87, LD010 + LS REQUIPED TO LOAD - Reliad5 OU_COG 22,49,57 @000mi_050 W87, LD010 + LS REQUIPED TO LOAD - Reliad5 OU_COG 22,49,57 @000mi_050 W87, LD010 + LS REQUIPED TO LOAD - Reliad5 OU_COG 22,49,54 @000mi_050 W87, LD010 + LS REQUIPED TO LOAD - Reliad5 OU_COG 22,49,40 @000mi_050 W87, L0000 LF,55 W72, 22,49,40 @000mi_051 W72, 23,49,40 @000mi_051 W72, 24,49,40 W700mi_051 W72, 25				
<pre>22,43,1000000,200 USA, .221 CP SECONDS COMPILATION TIME .22,43,5000000,220 USA, .221 CP SECONDS COMPILATION TIME .22,43,5000000,501 MFZ, UD010 = LIS REQUIPED TO LOAD - RW11445 OU_COG 22,43,57000000,501 WFZ, UD010 = LIS REQUIPED TO LOAD - RW11445 OU_COG 22,43,57000000,501 WFZ, UD010 = LIS REQUIPED TO LOAD - RW11445 OU_COG 22,43,57000000,505 USA, STUP 22,44,6000000,505 USA, STUP 22,44,6000000,505 USA, STUP 22,44,6000000,505 USA, STUP 22,44,6000000,505 USA, STUP 22,44,6000000,505 USA, STUP 22,44,600000000,507 WFZ, MAX M5 6 40K 11,507 CP SECONDS EXECUTION TIME. 22,44,6000000,507 WFZ, MAX M5 6 40K 11,000000,507 WFZ, JM160 - MAXIMUM USER SCM 1468078 MORDS 22,44,60000000,507 WFZ, JM160 - MAXIMUM USER SCM 1468078 MORDS 22,44,6000000000000000000000000000000000</pre>	22.49.31 00000.004	MFZ,	+ABOR2, STMFZ.	
22,49,50 UPUGU,229 LOU; -LGU. 22,49,50 UPUGU,257 MFZ; LDB13 + LS REUUIPEN TO LDAD + RW11445 NULCAG 22,49,57 UPUGUGU,50 USR; STUP 22,49,57 UPUGUGU,50 USR; STUP 22,49,40 UPUGUGU,669 USR; STUP 22,49,40 UPUGUGU,669 USR; 1,50T P SECONDS EXECUTION TIME. 22,49,40 UPUGUGU,669 USR; 1,50T P SECONDS EXECUTION TIME. 22,49,40 UPUGUGU,669 USR; 1,50T P SECONDS EXECUTION TIME. 22,49,40 UPUGUGU,669 USR; MAX M5 6 40KW 22,49,40 UPUGUGU,669 USR; MAX M5 6 40KW 22,49,40 UPUGUGU,672 MFZ; JM166 + NAXIMUN USER SCM 146407B WARDS 22,49,40 UPUGUGU,672 MFZ; JM167 + NAXIMUN USER SCM 146407B WARDS 22,49,40 UPUGUGU,672 MFZ; JM167 + NAXIMUN USER SCM 146407B WARDS 22,49,40 UPUGUGU,673 MFZ; SCN 75,966 KMS 22,49,40 UPUGUGUGU,673 MFZ; SCN 75,966 KMS 22,49,40 UPUGUGUGUSUSER CLC 8,4975 22,49,40 UPUGUGUGUSUSER CLC 8,4975 22,49,40 UPUGUGUGUSUSER CLC 8,4975 22,49,40 UPUGUGUGUSUSER CLC 8,4975 22,49,40 UPUGUGUGUSUSER CLC 8,4975 1 PPUGKAH IS30 76/76 0PT=2 FTN 4,845 1 PPUGUGUUSUSER CLC 8,4975 22,49,40 UPUGUGUSUSER CLC 8,4975 1 PPUGUGUGUSUSER CLC 8,4975 1 PPUGUGUGUSUSER CLC 8,4975 1 PPUGUGUSUSER CLC 8,4975 22,49,40 UPUGUSUSER CLC 8,4975 1 PPUGUGUSUSER CLC 8,4975 1 PPUGUSUSER CLC 8,4975 22,49,40 UPUSER CLC 8,4975 22,49,40 UPUSER CLC 8,4975 1 PPUGUSUSER CLC 8,4975 1 PPUGUSUSER CLC 8,4975 1 PPUGUSUSER CLC 8,4975 1 PPUGUSUSER CLC 8,4975 1 PPUSER CLC 8,4755 2 PUSER CLC 8,4757 2 PUSER CLC 8,47577 2 PUSER	22,49,31 00000,000	LOU.	<pre>#FTN.UPT#2.</pre>	
22.44,37         000003,561         MPZ,           22.44,37         000003,562         STUP           22.44,37         000003,562         STUP           22.44,42         000003,562         STUP           22.44,42         000003,562         STUP           22.44,44         00001,564         STUP           22.44,44         00001,564         STUP           22.44,44         00001,571         MEZ,           22.44,44         00001,572         MEZ,           22.44,44         00001,572         MEZ,           22.44,44         00001,573         MEZ,           22.44,44         00001,574         MEZ,           22.44,44         00001,574         MEZ,           23.449,44         00001,574         MEZ,           24.44,44         0001,574	22,49,34 00000,229 22,49,35 00000,357	MFZ.	-LOU. LODIN - FLS REQUIRED TO LOAD - RM11445 OU.COG	
22,49,40 40901,669 USR, STUP 22,49,40 40901,669 USH, 140400 FINAL EXECUTION FL. 22,49,40 40901,670 MFZ, MAX MS 6 *44W 22,49,40 40901,671 MFZ, JM160 = MAXIMUM USER LCM 340608 WORDS 22,49,40 40901,672 MFZ, JM167 = MAXIMUM USER LCM 340608 WORDS 22,49,40 40901,672 MFZ, JM167 = MAXIMUM USER LCM 340608 WORDS 22,49,40 40901,673 MFZ, SCh 75,896 KWS 22,49,40 40901,673 MFZ, LCH 7,806 KWS 22,49,40 40901,673 MFZ, LCH 7,806 KWS 22,49,40 40901,673 MFZ, LCH 7,806 KWS 22,49,40 40901,673 MFZ, JDU 7,100 1,441 SEC 22,49,40 40901,673 MFZ, JDU 1,441 SEC 22,49,40 40901,674 MFZ, JDU 1,441 SEC 22,49,40 40001,674 MFZ, JDU 1,464 SEC 22,49,40 40001,674 MFZ, SCG 974 SLLas 4 CALL RANSFT(1) SLL 22,49,40 40001,674 MFZ, SCG 974 SLLas 4 CALL RANSFT(1) SLL 1 PROGRAM IS30 76/76 OPT=2 FTN 4,845 1 PROGRAM IS30 76/76 OPT=2 FTN 4,845 1 PROGRAM IS30 76/76 OPT=2 FTN 4,845 1 PROGRAM IS30 16/74 MFZ, SCG 974 SLLas 4 CALL RANSFT(1) SLL=203LL 0 D 1 141,1 S INDM1(1)=10 1 INDM1(1)=1 INDP1(1)=1 1 INDM1(1)=1 S INDM1(1)=1 1 INDM1(1)=1 S INDM1(1) S JP1=INDP1(1) 0 D 4 KEL,1 S KM1=INDM1(K) S KP1=INDP1(K) 0 D 4 I=1,LL 1 INDM1(1)=1 S INDM1(1) S JP1=INDP1(1) 1 IEN=13(CL,J,K)/ST) S GOTO 12 1 IEN=13(CL,J,K)/ST) S GOTO 12 1 IEN=13(CL,J,K)/ST) S GOTO 12 1 IEN=13(CL,J,K)/ST) S GOTO 12 1 IEN=13(L,J,K)/ST) S GOTO 12 1 IEN=13(L,J,K)/ST) S GOTO 12 1 IEN=13(L,J,K)/ST) S GOTO 12 1 IEN=13(L,J,K)/ST) S GOTO 12 1 IEN=14(ST) S IEN=SHIFT(ICE),J,K)/ST) S GOTO 12 1 IEN=14(ST),J,K)/STS(L,J,K)/ST) S GOTO 12 1 IEN=14(ST),J,K)/STS(L,J,K)/ST) S GOTO 12 1 IEN=13(L,J,K)/ST) S GOTO 12	22.49.37 00000,361 22.49.37 00000,361	USR.	LD603 - EXECUTION INITIATED OS.EXP Fortran Library 528 - 25/05/81	
22,44,40 08081,669 USR,       1,307 CP SECONDS EXECUTION TTME.         22,44,40 08081,670 MFZ,       MAX MS       6 +4KU         22,44,40 08081,670 MFZ,       M166 - MAXIMUH USER SCM       146407B WRRNS         22,44,40 08081,672 MFZ,       M165 - MAXIMUH USER SCM       146407B WRRNS         22,44,40 08081,673 MFZ,       SCM       75.494 KWS         22,44,40 08081,673 MFZ,       SCM       75.494 KWS         22,44,40 08081,673 MFZ,       ICM       75.494 KWS         22,44,40 08081,673 MFZ,       ICM       75.494 KWS         22,44,40 08081,673 MFZ,       ICM       75.494 KWS         22,44,40 08081,673 MFZ,       USER       1.441 SEC         22,44,40 08081,673 MFZ,       USER       1.441 SEC         22,44,40 08081,674 MFZ,       SCMSU = 0808046 SC/LC SWAPS         PROGRAM ISSU       76/76 0PT=2       FTN 4.8*5         1       PPOGRAM ISSU (017917,14081(1807),1NDP1(1807)       UATA IS/S08048-07,1EN/1111111111111111111111111111         1       INDP1(1)=I       INDP1(1)=I       INDP1(1)=I         1       INPP(1)=14       INDP1(1)=I       INDP1(1)=I         1       INDP1(1)=I       INDP1(1)=I       INDP1(1)         1       INDP1(1)=I       INDP1(1)=I       INDP1(1)=I         0	22,49,40 00001,669	USR, USR,		
22,49,40 00001,670 HFZ, MAX MS 6 4000 22,49,40 00001,672 HFZ, MIST = MAXIMUH USER SCM 146400B WRRNS 22,49,40 00001,673 HFZ, JM167 = MAXIMUH USER SCM 1768 0UFFERS 22,49,40 00001,673 HFZ, SCH 75,896 KWS 22,49,40 00001,673 HFZ, ICH 7,7860 KWS 22,49,40 00001,674 HFZ, JOB 1,675 SFC 22,49,40 00001,674 HFZ, SCH 7, JOB 1,675 SFC 22,49,40 00001,675 HFZ, SCH 7, JOB 1,775 SFC 1 INDPI(1)1=1 ST 1,1 STNO11(1)=1-1 INDPI(1)101 IS 1,1 STNO11(1)=1-1 INDPI(1)101 IS 1,1 STNO11(1)=1-1 INDPI(1)101 IS 1,1 STNO11(1)=1-1 INDPI(1)100 4 STNO11(1)=1-1 INDPI(1) ID 0 4 STNO11(1) STNO11(1) STNO11(1) IS 1,1 STNO11(	22,49,40 00001,669	USR,	1,307 CP SECONDS EXECUTION TIME. +EXIT,5.	
22,49,40 00,001,672 HFZ, JM157 = MAXIMUM USER LCM 340,00 MARDS 22,49,40 00,001,673 HFZ, SCH 75,896 KHS 22,49,40 00,001,673 HFZ, LCH 75,896 KHS 22,49,40 00,001,673 HFZ, LSR 1.441 SEC 22,49,40 00,001,673 HFZ, JOU 85R 1.441 SEC 22,49,40 00,001,674 HFZ, SCHSU 90,0006 SC/LC SWAPS PHOURAM IS3U 76/76 OPT=2 FTN 4.8+5 PHOURAM IS3U 76/76 OPT=2 FTN 4.8+5 1 PPOGRAM IS3D (0UTPUT, TAPE5=0UTPUT) UTHERSION 18(5,100+10/,111111111111111111111 5 UD 141,157,100+10,101,11111111111111111111 5 UD 14,157,100+10,101,11111111111111111111 1 INDP1(1)=1 S INDP1(L)=1 0 0 Z 1=1,7 2 EX(I)=EXP((a=1=16)/T) 10 UA 4,1,L S HAPI(I) S JP1=INDP1(J) 00 4 Ja1,L S JM10HI(S) LS JP1=INDP1(J) 10 0 4 Ja1,L S JM10HI(S) LS JP1=INDP1(J) 11 IEN=15(LH1,JK)+SIS(I+I,JK) S GUTO 12 11 IEN=15(LH1,JK)+SIS(I+I,JK),ST) S GUTO 12 12 ICI=15(I,JK) 14 ICI=15(I,JK) 15 ICI=15(I,JK) 16 ICH=15(I,JK) S IIF1,I0(I,J,K),ST) S GUTO 12 11 IEN=15(LH1,JK)+SIS(I+I,JK),ST) S GUTO 12 12 ICI=15(I,JK) 14 ICI=15(I,JK) 15 ICI=15(I,JK) S IIF1,I0(I,J,K),ST) S GUTO 12 15 ICI=15(I,JK) 26 JIE1,ID 27 ICI=15(I,JK) S IIENSHIFT(ICI,J,K),ST) S GUTO 12 16 ICI=15(I,JK) S IIENSHIFT(ICI,J,K),ST) S GUTO 12 17 ICI=15(I,JK) S IIENSHIFT(ICI,J,K),ST) ICI=1C(I,J,K),ST) 26 J ICI=15(I,JK) S IIENSHIFT(ICI,JK),ST) ICI=1C(I,J,K),ST) 27 ICI=15(I,JK) S IIENSHIFT(ICI,JK),ST) ICI=1C(I,J,K),ST) 28 ICI=15(I,JK) S IIENSHIFT(ICI,JK),ST) ICI=1C(I,J,K),ST) 29 ICI=15(I,JK) S IIENSHIFT(ICI,JK),ST) ICI=1C(I,J,K),ST) 20 ICI=15(I,JK) S IIENSHIFT(ICI,JK),ST) ICI=1C(I,J,K),ST) 21 ICI=15(I,JK) S IIENSHIFT(ICI,JK),ST) ICI=1C(I,J,K),ST) 22 ICI=15(I,JK) S IIENSHIFT(ICI,JK),ST) ICI=1C(I,JK),ST) 23 IF PARAFERSEN 24 ICI=15(I,JK) S IIENSHIFT(ICI,JK),S	22,49,40 00001,670	MFZ, MFZ,	MAX MS	
22,49,40       00001,073 MFZ, 22,49,40       1/1       7.22 KW3         22,49,40       00001,073 MFZ, 22,49,40       1/1       0,001 MW         22,49,40       00001,073 MFZ, 22,49,40       USER       1,441       SEC         22,49,40       00001,073 MFZ, 360,40       00001,073 MFZ, 360,50       USER       1,441       SEC         22,49,40       00001,073 MFZ, 360,50       JOB       1,675       SFC         22,49,40       00001,074       MFZ, 370,221673       MFZ       FTN 4,045         1       PROGRAM IS30       76/76       OPT=2       FTN 4,045         1       DINERSION IS(5,100,100),EX(7),INDM;1100),INDP;1100       Otta;       IL         1       DINERSION IS(5,100,100),EX(7),INDM;1100),INDP;1100       Otta;       IL         1       INDP;113       IS(5,100,100),EX(7),INDM;1100),INDP;1100       IL         2       0       IF1,15       INDP;111111111111111111111111111111111111	22.49.40 00001.672	MFZ, MFZ,	JM170 - HAXIMUH JS+ID LCM 1768 BUFFERS	
<pre>22,49.40 00001,073 MFZ, USER 1,441 SEC 22,49.40 00001,074 MFZ, JOU 1,675 SFC 22,49.40 00001,074 MFZ, SCU5U = DR0006 SC/LC SMAPS PROGRAM ISSU 76/76 OPT=2 FTN 4,845 1 PROGRAM ISSU 76/76 OPT=2 FTN 4,845 1 PROGRAM ISSU 76/76 OPT=2 FTN 4,845 1 PROGRAM ISSU 76/76 OPT=2 FTN 4,845 5 DO 1 Te1,12 STA00,000,TEN1/11111111111111111111111111111111111</pre>	22,49,40 00001,673	MFZ.	LCH 2,722 KWS	
<pre>22.49.40 00001.674 HFZ. SC050 = 000006 SC/LC SWAPS PROGRAM ISSU 76/76 OPT=2 FTN 4.8+5 1 PPOGRAM ISSU 76/76 OPT=2 FTN 4.8+5 1 PPOGRAM ISSU 76/76 OPT=2 FTN 4.8+5 1 PPOGRAM ISSU 76/76 OPT=2 FTN 4.8+5 1 UNENSION 18(5,100,100,EX(7),INDM1(100),INDP1(100) 0474 IS/50000+07,IEN1/11111111111111111111111111111 5 D0 1 I=1,L S INDM1(I)=I=1 1 INDP1(I)=L S INDP1(L)=1 D0 2 I=1,7 2 EEX(1)=2EYP((a=1=6)/T) 10 0 4 K=1,L S INDP1(L)=1 D0 4 K=1,L S INDP1(L)=1 D0 4 K=1,L S INDP1(L) S OFTO 11 15 IEN=13(I=1,J,K)+S(I=1,J,K) S GOTO 12 16 IEN=13(I=1,J,K)+S(I=1,J,K) S GOTO 12 17 IEN=13(I=1,J,K)+S(I=1,J,K),S) S GOTO 12 18 IEN=13(I=1,J,K)+SHIFT(IS(1,J,K),S) 19 IEN=13(I=1,J,K)+SHIFT(IS(1,J,K),S) 10 IEN=13(I=1,J,K)+SHIFT(IS(1,J,K),S) 12 ICI=13(I,J,K) 14 ICI=13(I,J,K) 15 IF(R=RNF(ICI-N,SHIFT(ICI,1),OR,SHIFT(ICI,2),IEN) 16 ICI=55(I,J,K) S IEN=SHIFT(IEN,3) 17 (RANF(II),LE,EX(IEN,AND,7)) ICH=ICH,OP,7 16 IEN=SHIFT(ICH,3) S IEN=SHIFT(IEN,3) 17 (RANF(II),LE,EX(IEN,AND,7)) ICH=ICH,OP,7 4 ICI=13(I,J,K)&gt;COM(ICI,ICH,AND,7)) ICH=ICH,OP,7 5 FORMAT(2I20) 3TUP 5 END 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1</pre>	22,49,40 00001,67	MFZ.	USER 1,441 SEC	
<pre>1</pre>	22,49,40 00001,674 22,49,40 00001,674	MFZ	ЈОВ	
<pre>DIMENSION is(5,100,100,12(7),NDM1(100),NDP1(100) UATA IS/50000e0/,IEN1/11111111111111111111111111 5 UD 1 I=1,L S INDM1(1)=I=1 INDP1(1)=L; INDP1(L)=1 DO 2 I=1,7 2 EX(1)=EYP((a=1=16)/T) DO 4 I=1,L S HN=INDM1(K) S RP1=INDP1(K) DO 4 I=1,L S HM=INDM1(J) S JP1=INDP1(K) DO 4 I=1,L S HM=INDM1(J) S JP1=INDP1(J) DO 4 I=1,L S GUTO 10 S IF(I,EG,LL) GOTO 11 IEN=I3(I=1,J,K)+IS(I+1,J,K) S GUTO 12 II IEN=I3(I=1,J,K)+IS(I+1,J,K),S GUTO 12 II IEN=I3(I=1,J,K)+IS(I,J,K),S) I2 ICI=I3(I,J,K) I2 ICI=I3(I,J,K) I2 ICI=I3(I,J,K) I2 ICI=I3(I,J,K) I2 ICI=I3(I,J,K) I2 ICI=I3(I,J,K) I2 ICI=I3(I,J,K) I3 IF(RANF(II),LE,EX(IEN,AND,T)) ICH=ICH,OP,T ICH=SHIFT(ICH,S) S IEN=SHIFT(IEN,S) I3 IF(RANF(II),LE,EX(IEN,AND,T)) ICH=ICH,OP,T ICH=SHIFT(ICH,J,K)+ITME S FORMAT(2I20) STUP S END 0 ICH 0 ICH 0</pre>				
UATA IS/50000407,IEN1/11111111111111111111111111111111111	PROGR	M IS30	76/76 0PT≡2 FTN 4.8+	5
<pre>5</pre>		M IS3D	PPOGRAM IS3D(DUTPUT,TAPE5=DUTPUT)	5
<pre>INDM1(1)=L S INDP1(L)=1 D0 2 I=1,7 2 EX(I)=EXP((d+I=16)/T) 10 0 f ITIME=1,6 \$ M=0 D0 4 K=1,L \$ KHI=INDM1(K) \$ KPI=INUP1(K) D0 4 J=1,L \$ KHI=INDM1(J) \$ JPI=INDP1(J) D0 4 I=1,LL IF(I,EU,I) GOTO 10 \$ IF(I_EU,LL) GOTO 11 IS IEN=I3(I=1,J,K)*IS(I+1,J,K) \$ GUTO 12 11 IEN=I3(L,J,K)*IS(I+1,J,K), \$ GUTO 12 11 IEN=I3(L,J,K)*IF(IS(L,J,K),57) \$ GOTO 12 11 IEN=I3(L,J,K)*SHIFT(IS(L,J,K),57) 12 ICI=I3(I,J,K) IF(N=ICN+ICI=IS(I,JM1,K)+IS(I,JP1,K)*IS(I,J,KM1)+I5(I,J, IF(N=ICN+ICI=IS(I,JM1,K)+IS(I,J),CR_SHIFT(ICI,2),TEN) ICH=USUIFT(ICH,3) \$ JEN=SHIFT(TEN,3) IF(RANF(II)_LE_EX(IEN_AND,7)) ICH=ICH,OP,7 ICH=USHIFT(ICH,3) \$ JEN=SHIFT(TEN,3) IF(RANF(II)_LE_EX(IEN_AND,7)) ICH=ICH,OP,7 4 ICI=I3(I,J,K) 5 FORMAT(2120) 3 TOP \$ END 0 1 0 2 0 1 0 3 0 4 0 5 0 4 0 5 0 5 0 1 0 5 0 5 0 5 0 1 0 5 0 1 0 5 0 1 0 5 0 1 0 5 0 1 0 5 0 1 0 1 0 5 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1</pre>		M IS30	PROGRAM IS3D(DUTPUT,TAPE5=00JTPUT) UIHERAJON 18(5,100/100),EX(7),INDM1(100),TNDP1(100) UATA IS/50004040/,IEN/111111111111111111111	
<pre>2 EX(1)=EXP((a=1=(b)/T) 10 D0 6 TITHE=1:6 M=0 D0 6 TITHE=1:6 M=0 D0 4 K=1;L 5 KH1=INDM1(K) 5 KP1=INDP1(K) D0 4 J=1;L 7 JH1=INDM1(J) 5 JP1==NDP1(J) D0 4 J=1;L 7 IF(I;E0,LL) 6 0TO 11 IF(I;E0,L) 5 (0TO 10 5 IF(I;E0,LL) 6 0TO 12 IF(I;E1,J,K) + ST(I+1,J,K) 5 0TO 12 If IEN=13(2,J,K) + SHIFT(IS(L,J,K),57) 5 0TO 12 If IEN=13(2,J,K) + SHIFT(IS(L,J,K),57) 5 0TO 12 If IEN=13(L,J,K) IEN=15(L,J,K) IEN=15(L,J,K) IEN=15(L,J,K) IEN=15(L,J,K) IEN=15(I,J,K) IEN=15(I,J,K) IEN=15(I,J,K) IEN=15(I,J,K) IEN=15(I,J,K) IEN=15(I,J,K) IEN=15(I,J,K) IEN=15(I,J,K) IF(RANF(II),LE,EX(IEN,AND,T)) IEN=16H,0P,7 ICH=05 D0 3 IEN=15HFF(IEN,3) IF(RANF(II),LE,EX(IEN,AND,T)) IEN=16H,0P,7 4 ICI=13(I,J,K)=X0F(ICI,ICH,AND,IENI) 6 WFITE(5:5) M,ITIME 5 FORMAT(2120) STUP 5 END 0 1 0 2 0 4 0 5 0 4 0 5 0 5 0 5 0 5 0 5 0 5 0 5 0 5 0 5 0 5</pre>	1	IN IS30	PPOGRAM IS3D(DUTPUT,TAPE5=00TPUT) DIMENSION IS(5,100,100),EX(7),INDM1(100),TNDP1(100) OATA IS/50000±07,IEN/1111111111111111111111 T=1,27923/0,221673 \$ LL=3 \$ CALL RANSET(1) \$ L=LL*20SL D0 1 I=1,L \$ INDM1(1)=I=1	
D0 4 Kei,L S KHISINDMI(K) S RPJSINDP1(K) D0 4 Jei,L S JMISINDMI(J) S JPISINDP1(J) D0 4 IEI,L IF(I,GU,I) GOTO 10 S IF(IEG,LL) GOTO 11 IF(I,GU,I) GOTO 10 S IF(IEG,LL) GOTO 12 I0 IENSIS(I=I,J,K) SIS(I=I,J,K) S GOTO 12 11 IENSIS(L,MI,J,K) SHIFT(IS(L,J,K),S) S GOTO 12 12 ICIS(I,J,K) IENSIS(I=IS(I,J,K) IENSIS(I=I,J,K)+IS(I,J,KMI)+IS(I,J,K)) IENSIS(I=KONF(ICIC,S) SIENSISTF(ICI,S),OR,SHIFT(ICI,Z),TEN) ICHSON JO S IISI,I0 ICHSON JO S IISI,I0 ICHSON JISI,I0 ICHSON JISI,I0 IF(RANF(II),LE,EX(IEN,AND,7)) ICHSICH,OP,7 ICHSON JISIS(I,J,K) SIONSISTF(ICI,S) STOP S END 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0	1		PROGRAM IS30(OUTPUT,TAPE5=OUTPUT) UTHENSION IS(5,100,100)EX(7),INDM((100),INDP((100) UATA IS/500000+07,IEN/1111111111111111111111 Fa1,27923/d,221673 \$ LL=3 \$ CALL RANSET(1) \$ L=LL*20SL DO 1 I=1,L \$ INOM1(I)=I=1 INUP1(I)=L \$ INOP1(L)=1	
D0 4 1=1,LL IF (1,E0,1) GOTO 10 S IF (1,E0,LL) GOTO 11 15 IEN=13(I=1,J,K)+IS(I=1,J,K) S GOTO 12 10 IEN=13(2,J,K)+SMIFF(IS(LL,J,K),T) S GOTO 12 11 IEN=13(L(LM1,J,K)+SMIFF(IS(L,J,K),T) S 12 ICI=13(I,J,K) 14 ICI=13(I,J,K) 16 IEN=1EN1+XOP(ICI.ON_SMIFF(ICI.1),OR_SMIFT(ICI.2),TEN) 17 ICI=0 & DO 3 II=1,10 ICH=SMIFF(ICH.3) S IEN=SMIFF(TEN,3) 17 (RANN (II)_LE_EX(IEN_AND_T)) ICH=ICH,OP_T 16 H=SMIFF(ICH.3) S IEN=SMIFF(TEN,3) 17 (RANN (II)_LE_EX(IEN_AND_T)) ICH=ICH.0P_T 4 ICI=13(I,J,K)=XOP(ICI.ICI.ANU,IENI) 5 YORMAT(2120) 3 TUP S END 0 1 0 2 1 2 1 3 0 4 0 5	1	1	<pre>PPOGRAM IS3D(DUTPUT,TAPE5=DUTPUT) DIMENSION IS(5,100/100),EX(7),INDM1(100),INDM1(100) OATA IS/S0000+07,IEN1/111111111111111111110 T=1,27923/0,221673 \$ LL=3 \$ CALL PANSET(1) \$ L=LL*20SL DO 1 T=1,L \$ INOM1(1)=I=1 INDM1(1)=L \$ INOM1(L)=1 DO 2 I=1,7 EX(1)=EXP((4=I=16)/T)</pre>	
<pre>IF(I,EQ.1) GOTO 10 3 IF(I,EQ.L.) GOTO 11 IF(I,EQ.1) GOTO 10 3 IF(I,EQ.L.) GOTO 12 IF(I=1,J,K)+SHIF(IS(I+1,J,K) S GOTO 12 IF(I=1) I(I=1,J,K)+SHIF(IS(I,J,K),57) S GOTO 12 IF(I=1) I(I=1) I(I,J,K)+IS(I,J,K),3) IC(I=13(I,J,K) IC(I=13(I,J,K) IC(I=13(I,J,K) IC(I=13(I,J,K) IC(I=13(I,J,K)) IC(I=13(I,J,K) IC(I=13(I,J,K)) IC(I=13(I,J,K) IC(I=13(I,J,K)) IC(I=13(I,J,K)) IC(I=13(I,J,K)) IC(I=13(I,I) IC(I=13(I,I)) IC(I=13(I,I,I)) IC(I=13(I,I,I)) IC(I=13(I,I,I)) IC(I=13(I,I,I)) IC(I=13(I,I,I)) IC(I=13(I,I,I)) IC(I=13(I,I,I)) IC(I=13(I,I,I)) IC(I=13(I,I,I)) IC(I=13(I,I)) IC(I=II) IC(I=III) IC(I=II) IC(II) IC(I</pre>	1	1	<pre>PPOGRAM IS3D(DUTPUT,TAPE5=DUTPUT) UTHENSION IS(5,100/100),EX(7),INDM1(100),INDM1(100) OATA IS/50000400/.EX(7),INDM1(101)1111111111111111 T=1,27923/U,221673 \$ LL=3 \$ CALL PANSET(1) \$ L=LL*203L DO 1 I=1,L \$ INDM1(1)=I=1 INDM1(1)=L \$ INDM1(L)=1 DO 2 I=1,7 EX(I)=EXP((4*I=16)/T) DO 4 ITIME=1,6 \$ M=0 DO 4 K=1,L \$ KM1=INDM1(K) \$ KM1=INDM1(K)</pre>	
11       IEN=I3(LLM1, J, K)+SHIFT(IS(1, J, K), 3)         12       ICI=I3(I, J, K)         14       ICI=I3(I, J, K)         15       ICI=I3(I, J, K)         16       IEN=IEN+ICI=I3(I, JM1, K)+IS(I, JP1, K)+IS(I, J, KM1)+I3(I, J, IEN=IEN+ICICI, I), OR, SHIFT(ICI, 2), TEN)         16       IEN=IEN+XOP(ICI, ON, SHIFT(ICI, I), OR, SHIFT(ICI, 2), TEN)         17       IEN=IEN+XOP(ICI, ON, SHIFT(ICI, I), OR, SHIFT(ICI, 2), TEN)         18       IEN=SHIFT(ICH, 3)         19       ICH=SHIFT(ICH, 3)         11       IENESHIFT(ICH, 3)         11       IENESHIFT(ICH, 3)         11       IENESHIFT(ICH, 3)         12       IF(RANF(II), LE_EX(IEN, AND, 7))         14       ICI=I3(I, J, K) = XORH(ICI, ICH, AND, 7)         15       FORMAT(2120)         16       WRITE(5,5)         17       I         18       I         19       I         10       I         11       I         12       I         13       I         14       I         15       FORMAT(2120)         16       I         17       I         18       I         19       I </td <th>1</th> <td>1</td> <td><pre>PPOGRAM IS3D(DUTPUT,TAPE5=DUTPUT) UTHERASION IS(5,100/100),EX(7),INDM1(100),INDP1(100) UATA IS/50000407,IEN/111111111111111111111111 T=1,27923/U,221673 % LL=3 % CALL PANSFT(1) % L=LL*20SL D0 1 I=1,L \$ INDM1(I)=I=1 INDM1(1)=L \$ INDM1(I)=I=1 INDM1(1)=L \$ INDP1(L)=1 D0 2 I=1,7 EX(I)=EXP((d=I=16)/T) D0 6 ITIME=1+6 \$ M=0 D0 4 K=1,L \$ KM1=INDM1(K) \$ KPI=INDP1(K) D0 4 J=1,L </pre></td> <td></td>	1	1	<pre>PPOGRAM IS3D(DUTPUT,TAPE5=DUTPUT) UTHERASION IS(5,100/100),EX(7),INDM1(100),INDP1(100) UATA IS/50000407,IEN/111111111111111111111111 T=1,27923/U,221673 % LL=3 % CALL PANSFT(1) % L=LL*20SL D0 1 I=1,L \$ INDM1(I)=I=1 INDM1(1)=L \$ INDM1(I)=I=1 INDM1(1)=L \$ INDP1(L)=1 D0 2 I=1,7 EX(I)=EXP((d=I=16)/T) D0 6 ITIME=1+6 \$ M=0 D0 4 K=1,L \$ KM1=INDM1(K) \$ KPI=INDP1(K) D0 4 J=1,L </pre>	
1EN#1EN#ICI+13(I,JM1,K)+15(I,JF(K)+13(I,J,KM1)+16(I,J,         20       1EN#1EN1+KOR(ICI,OM_SHIFT(ICI,1),OR,SHIFT(ICI,2),TEN)         1E(H=0 & DO 3   I=1,10         25       3   F(RANF(II)_LE_EX(IEN,AND,7))   ICH=ICH,0R,7         1E(I=10(I)_J,K)=XEOR(ICI,ICH,ANU,IEN1)         6       WRITE(5,5) M,ITIME         5       FORMAT(2120)         3TUP S END         0       1         0       5	1 5 1@	1	<pre>PPOGRAM IS3D(DUTPUT,TAPE5=DUTPUT) UTHERASION IS(5,100,100),EX(7),INDM1(100),INDP1(100) UATA IS/50004040,IEN(7)11111111111111111111111 Fa1,27923/U,221673 % LL=3 &amp; CALL PANSFT(1) % L=LL+20%L D0 1 I=1,L \$ INDM1(1)=I=1 INDM1(1)=L \$ INDM1(L)=1 D0 2 I=1,7 EX(1)=EXP((d=I=6)/T) D0 6 ITHE=1,6 \$ M=0 D0 4 K=1,L \$ KH1=INDM1(K) \$ KP1=INDP1(K) D0 4 J=1,L IF(I,EQ,I) &amp; GOTO 10 IF(I,EQ,I) &amp; GOTO 11 IEN=15(L II-1,JK) \$ GOTO 12 </pre>	
ICH=0 & DO 3 II=1,10 ICH=SMIFT(ICH,3) & IENEBHFFT(IEN,3) IF(RANK[II]_LE_EX(IEN,AND,7)) ICH=ICH,0P,7 ICHESMIFT(ICH,3) & IENEBHFFT(JEN,3) 25 3 IF(RANF(II],LE_EX(IEN,AND,7)) ICH=ICH,0P,7 4 ICI=13(I,J,K)=XOM(ICI,ICH,AND,IENI) 6 WRITE(5,5) M,ITIME 5 FORMAT(2120) 3 STUP S END 0 1 0 2 1 0 3 0 4 0 5	1 5 1@	1 2 10 11	<pre>PPOGRAM IS3D(DUTPUT,TAPE5=DUTPUT) DIMENSION IS(5,100/100),EX(7),INDM1(100),INDM1(100) UATA IS/50000+07.IEN1/1111111111111111111111 T=1,27923/U,221673 % LL=3 % CALL PANSET(1) % L=LL*20%L DO 1 I=1,L \$ INDM1(1)=I=1 INDM1(1)=L \$ INDM1(L)=1 DO 2 I=1,7 EX(I)=EXP((4=I=16)/T) DO 4 K=1,L \$ MHI=INOM1(K) \$ KPI=INDP1(K) DO 4 J=1,L \$ JMI=INOM1(K) \$ KPI=INDP1(J) DO 4 J=1,L \$ JMI=INOM1(J) \$ JPI=INDP1(J) DO 4 I=1,LL IF(I,E0,L) \$ GOTO 10 \$ IF(I.E0,LL) \$ GOTO 11 IEN=18(I=1,J,K)+\$SI[F1(IS(L,J,K),57) \$ GOTO 12 IEN=13(LLM1,J,K)+\$SHIFT(IS(L,J,K),3)</pre>	
IF(RANF(II), LE, EX(IEN, ANU, 7)) ICH=ICH, OP, 7 ICH=SMIFT(ICH, 3) S IENSBHTF(TEN, 3) 25 3 IF(RANF(II), LE, EX(IEN, AND, 7)) ICH=ICH, OP, 7 4 ICI=13(I,J,X)=XOP(ICI,ICH, AND, IEN1) 6 WRITE(5,5) M, ITIME 5 FORMAT(2120) 3TUP 5 END 0 1 8 2 0 3 0 4 0 5	1 5 10 15	1 2 10 11	<pre>PPOGRAM IS3D(DUTPUT,TAPE5=DUTPUT) UTHENSION IS(5,100,100),EX(7),INDM1(100),INDM1(100) UTATA IS/50004-04,IEN/111111111111111111111111111111111111</pre>	L
25 3 IF(RANF(II)_LE_EX(IEN,AND,7)) ICHWICH,OR,7 4 ICINIS(I,J,K)=XOPK(ICI,ICH,ANU,IEN1) 6 WRITE(5,5) M,ITIME 5 FORMAT(2120) 8 10 0 1 0 2 0 3 0 4 0 5	1 5 10 15	1 2 10 11	<pre>PPOGRAM IS3D(DUTPUT,TAPE5=DUTPUT) UTHENSION IS(5,100,100),EX(7),INDM1(100),INDM1(100) OATA IS/50000407,IEN/11111111111111111111111111 T=1,27923/U,221673 % LL=3 &amp; CALL PANSFT(1) % L=LL+20%L D0 1 I=1,L % INDM1(I)=I=1 INDM1(1)=I+1 INDM1(1)=I+1 INDM1(1)=I+1 D0 2 I=1,7 EX(I)=EXP((a+I=16)/T) D0 4 I=1,L % HH=INDM1(K) % KPI=INDP1(K) D0 4 J=1,L % HH=INDM1(K) % KPI=INDP1(J) D0 4 J=1,L % JHI=INDM1(K) % SIF(I,E0,LL) % GOTO 11 IEN=15(L,J,K)*SHIF(I,E(L,J,K),57) % GOTO 12 IEN=13(LLM1,J,K)*SHIFT(IS(I,J,K),3) ICI=15(I,J,K) IEN=IEN+ICI+IS(I,JM1,K)+IS(I,JP1,K)+IS(I,J,KM1)+IS(I,J) IEN=1EN+X0(ICI,ON,SHIFT(ICI,I).OR,SHIFT(ICI,2),TEN)</pre>	L
6 WRITE(5,5) M,ITIME 5 FORMAT(2120) 0 Stup S End 0 R 0 R 0 S 0 S	1 5 10 15	1 2 10 11	<pre>PPOGRAM IS3D(DUTPUT,TAPE5=DUTPUT) UTHERSION IS(5,100,100),EX(7),INDM1(100),INDP1(100) UATA IS/50000407,IEN/11111111111111111111111111 Ts1,27923/0,221673 % LL=3 &amp; CALL PANSFT(1) % L=LL+20%L D0 1 T=1,L % INDM1(1)=T=1 INDM1(1)=L % INDM1(L)=1 D0 2 T=1,T EX(1)=EXP((d=T=16)/T) D0 6 ITIME=1,6 % M=0 D0 4 J=1,L % JM1=INDM1(K) % KP1=INDP1(K) D0 4 J=1,L % JM1=INDM1(K) % KP1=INDP1(J) D0 4 J=1,L IF(1,F0,1) &amp; GOTO 10 % IF(1,E0,LL) &amp; GOTO 11 IEN=15(L=1,J,K) % GOTO 12 IEN=15(2,J,K)+S0(T+1,J,K) % GOTO 12 IEN=15(2,J,K)+S0IFT(IS(1,J,K),3) IF1=15(T,J,K) IFN=IEN+TCIE1\$(T,JM1,K)+IS(T,JF1,K)+IS(T,J,KM1)+I\$(T,J IEN=15(LLM1,J,K) % IEN&amp;SHIFT(ICI,1).0R,SHIFT(ICI,2),TEN) ICH=80 &amp; D0 3 II=1,10 ICH=80 &amp; D0 3 II=1,10 ICH=80 &amp; ICH(1)_LEX(IEN,AND,7)) ICH=1CH_0P,7</pre>	L
ີ 3TUP SÎENDÎ 20 1 20 2 0 2 0 3 0 4 0 5	1 5 10 15 20	1 2 10 11 12 3	<pre>PPOGRAM IS3D(DUTPUT,TAPE5=DUTPUT) UTHERSIGN IS(5,100,100,EX(7),INDM1(100),INDP1(100) GATA IS/50004007,IEN/1111111111111111111111111 Te1,27923/U,221673 % LL=3 &amp; CALL PANSET(1) % L=LL+20%L D0 1 [=1,L \$ INDM1(1)=I=1 INDM1(1)=L \$ INDM1(L)=1 D0 2 [=1,7 EX(1)=EXP((d=I=16)/T) D0 6 ITHE=1.6 &amp; M=0 D0 4 K=1,L \$ KM1=INDM1(K) \$ KP1=INDP1(K) D0 4 ITHE=1.6 &amp; M=0 D0 4 K=1,L \$ KM1=INDM1(K) \$ JP1=INDP1(J) D0 4 ITHE=1.6 \$ M=0 D0 4 I=1,L IF(I,E0,1) &amp; GOTO 10 \$ IF(I,E0,LL) &amp; GOTO 11 IEN=18(L=1,J,K) \$ GOTO 12 IEN=13(2,J,K)+SM1FF(IS(L,J,K),57) \$ GOTO 12 IEN=13(2,J,K)+SM1FF(IS(L,J,K),3) IE1=13(1,J,K) IEN=1E1,J,K) IEN=1E1+XOP(ICI,ON_SM1FF(ICI,1),CR=SM1FF(ICI,2),TEN) ICH=0 \$ DO 3 II=1,10 ICH=8M1FF(ICH-3) \$ IEN=8M1FF(IEN,3) IF(RAMF(II)_LE_EX(IEN_AMD,7)) ICH=1CH_0P,7 ICH=5M1FF(ICH,3) \$ IEN=8M1FF(ICEN,3) IF(RAMF(II)_LE_EX(IEN_AMD,7)) ICH=1CH_0P,7</pre>	L
8 2 7 3 8 4 7 5	1 5 10 15 20	1 2 10 11 12 3 4 6	<pre>PPOGRAM ISJD(DUTPUT,TAPE5=DUTPUT) UTHERSIGN IS(5,100,100,EX(7),INDM1(100),INDP1(100) UTATA IS/50000407,IEM/11111111111111111111111111 T=1,27923/U,221673 % LL=3 &amp; CALL PANSFT(1) % L=LL+20%L D0 1 I=1,L \$ INDM1(1)=I=1 INDP1(1)#I=1 INDP1(1)#I=1 INDP1(1)#I=1 D0 2 I=1,7 EX(1)#EXP((d=I=16)/T) D0 6 ITHE=1,6 % H=0 D0 4 K=1,L \$ KH1=INDM1(K) \$ KP1=INDP1(K) D0 6 ITHE=1,6 % H=0 D0 4 K=1,L \$ KH1=INDM1(K) \$ KP1=INDP1(K) D0 4 J=1,L IF(I=Gu_1) GCTO 10 \$ IF(I=GLL) &amp; GCTO 11 IEN=15(2,J,K)+SH1FT(IS(L,J,K),57) \$ GCTO 12 IEN=15(I,M1,J,K)+SH1FT(IS(I,J,K),57) \$ GCTO 12 IEN=15(I,J,K) IEN=15(I,J,K) IEN=16(LM1,J,K)+SH1FT(IC1(J,J),CR,SH1FT(IC1,2),TEN) ICH=15(I,J,K) \$ IEN=8H1FT(IEN,3) IF(RANF(II),LE,EX(IEN_AND,7)) ICH=ICH,0P,7 ICH=SH1FT(ICH,3) \$ IEN=8H1FT(IEN,3) IF(RANF(II),LE,EX(IEN,AND,7)) ICH=1CH,0P,7 ICH=15(I,J,K)=XOP(IC1,ICH,AND,7)) ICH=1CH,0P,7 ICH=15(I,J,K)=XOP(IC1,ICH,AND,7) ICH=1CH,0P,7 ICH=15(I,J,K)=XOP(IC1,ICH,AND,7) ICH=1CH,0P,7 ICH=15(I,J,K)=XOP(IC1,ICH,AND,7) ICH=1CH,0P,7 ICH=15(I,J,K)=XOP(IC1,ICH,AND,7) ICH=1CH,0P,7 ICH=15(I,J,K)=XOP(IC1,ICH,AND,7) ICH=1CH,0P,7 ICH=15(I,J,K)=XOP(IC1,IC1,AND,7) ICH=1CH,0P,7 ICH=15(I,J,K)=XOP(IC1,IC1,AND,7) ICH=1CH,0P,7 ICH=15(I,J,K)=XOP(IC1,IC1,AND,7) ICH=1CH,0P,7 IC1=15(I,J,K)=XOP(IC1,IC1,AND,7) ICH=1CH,0P,7 IC1=15(I,J,</pre>	L
0 4 0 5	1 5 10 15 20	1 2 10 11 12 3 4 6 5	<pre>PPOGRAM IS3D(DUTPUT,TAPE5=DUTPUT) UTHENSION IS(5,100,100,EX(7),INDM1(100),INDM1(100) UATA IS/50004004,IEN(1)11111111111111111111111111111111111</pre>	L
	1 5 10 15 20	1 2 10 11 12 3 4 6 5 8 8	<pre>PPOGRAM IS3D(DUTPUT,TAPE5=DUTPUT) UTHENSION IS(5,100,100),EX(7),INDM1(100),INDM1(100) UATA IS/50000407,IEM/111111111111111111111111111111111111</pre>	L
0 6	1 5 10 15 20	1 2 10 11 12 3 4 5 5 0 0 0 0 0	<pre>PPOGRAM IS3D(DUTPUT,TAPE5=DUTPUT) UTHENSIGN IS(5,100,100),EX(7),INDM1(100),INDP1(100) GATA IS/50000407,IEN/1111111111111111111111111111 T11,27923/0,221673 % LL=3 &amp; CALL PANSFT(1) % L=LL+20%L D0 1 T=1,L % INDM1(1)=I=1 INDM1(1)=L &amp; INDM1(L)=1 D0 2 I=1,7 EX(I)=EXP((d=I=16)/T) D0 4 I=1,L % Mt=INDM1(K) % KPI=INDP1(K) D0 4 J=1,L % JHI=INDM1(J) % JPI=INDP1(J) D0 4 J=1,L % JHI=INDM1(J) % JPI=INDP1(J) D0 4 J=1,L % INDM1(L)L,L,L,L % GOTO 11 IEN=15(L=1,J,K)+IS(J+1,J,K) % GOTO 12 IEN=13(LLM1,J,K)+SHIFT(IS(1,J,K),3) ICI=15(I,J,K) IF(N=IEN+ICI+IS(I,JM1,K)+IS(I,J),R,SHIFT(ICI,2),TEN) ICH=0 D0 3 II=1,10 IF(RANF(II)_LE,EX(IEN,AND,7)) ICH=ICH,0P,7 ICH=0 D0 3 II=1,10 IF(RANF(II)_LE,EX(IEN,AND,7)) ICH=ICH,0P,7 ICH=0 SHIFT(ICH,0) IF(RANF(II)_LE,EX(IEN,AND,7)) ICH=1CH,0P,7 ICI=15(I,J,K)=XOH(ICI,ICI,ICH,AND,IEN) IF(RANF(II)_LE,EX(IEN,AND,7)) ICH=1CH,0P,7 ICI=15(I,J,K)=XOH(ICI,ICI,ICH,AND,IEN) IF(RANF(II)_IEN) IF(RANF(II)_IEND(IEN) IF(RANF(II)_IEND(IEN) IF(RANF(II)_IEND(IEN) IF(RANF(II)_IEND(IEND(IEN) IF(RANF(II)_IEND(IEND(IEND) IF(RANF(II)_IEND(IEND(IEND) IF(RANF(II)_IEND(IEND(IEND) IF(RANF(II)_IEND(IEND(IEND) IF(RANF(II)_IEND(IEND(IEND) IF(RANF(II)_IEND(IEND(IEND) IF(RANF(II)_IEND(IEND(IEND) IF(RANF(II)_IEND(IEND(IEND) IF(R</pre>	L

\*A800230 AF DC=40 FS=0001

Fig. 2. A complete program which ran on our Cyber 76 to test computing times of multispin coding in three-dimensional Ising models up to  $100 \times 100 \times 100 \times 100$ . It does not calculate *E* and *M* since that should be done at the end only.

#### ACKNOWLEDGMENTS

We thank D. Stauffer for suggestions and help with the manuscript, and H. G. Baumgaertel for a copy of his computer program of Ref. 4.

### REFERENCES

- 1. K. Binder, Monte Carlo Methods in Statistical Physics (Springer Verlag, Heidelberg, 1979).
- M. Creutz, L. Jacobs, and C. Rebbi, *Phys. Rev. Lett.* 42:1390 (1979); R. Zorn, H. J. Herrmann, and C. Rebbi, *Computer Phys. Commun.* 23:337 (1981); See also R. Friedberg and J. E. Cameron, *J. Chem. Phys.* 52:6049 (1970).
- 3. H. Ottavi, Z. Phys. B 44:203 (1981).
- 4. B. K. Chakrabarti, H. G. Baumgaertel, and D. Stauffer, Z. Phys. B 44:333 (1981).
- 5. D. Stauffer, Private Communication.