MONTE-CARLO SIMULATION OF PURE $U(N)$ AND SU(N) LATTICE GAUGE THEORIES WITH FUNDAMENTAL AND ADJOINT COUPLINGS *

R.W.B. ARDILL, K.J.M. MORIARTY

Department of Mathematics, Royal Holloway College, Englefield Green, Egham, Surrey TW20 0EX, UK

and

Michael CREUTZ

Brookhaven National Laboratory, Upton, NY 11973, USA

Received 5 July 1982

PROGRAM SUMMARY

Title of program: SUUNFA

Catalogue number: AAOT

Program available from: CPC Program Library, Queen's University of Belfast, N. Ireland (see application form in this issue)

Computer: CDC 6600, CDC 7600; *Installation*: University of London Computer Centre

Operating system: CDC NOS/BE, SCOPE

Programming language: FORTRAN IV with a few CDC non-standard features

High speed storage required: 26 K (maximum)

Number of bits in a word: 60

Peripherals used: card reader, line printer

Number of cards in combined program and test deck: 852

Card punching code: CDC

Keywords: lattice gauge theory, $U(N)$, SU(N), $U(N)/Z_N$ and SU(N)$/Z_N$ gauge theories, fundamental and adjoint representations, Yang–Mills theory, Abelian- and non-Abelian gauge theories, QED and QCD models, non-perturbative effects, phase transitions, confining and deconfining phases, quark theory, statistical mechanical analogies, action per plaquette, Metropolis algorithm, Monte Carlo techniques

Nature of the physical problem
The program simulates thermal equilibrium for $U(N)$ and SU(N) lattice gauge theories with couplings in both the fundamental and adjoint representations. Gauge theories on a lattice were originally proposed by Wilson [1] and Polyakov [2].

Method of solution
A Monte Carlo simulation of the system set up on a lattice of variable dimensionality and lattice size generates a sequence of field configurations on the lattice links. The Metropolis algorithm [3], originally developed for Monte Carlo simulations in statistical mechanics, is used to generate statistical equilibrium. New configurations are generated link by link and convergence to equilibrium is accelerated by performing the Metropolis algorithm NTMAX times on a given link before passing to the next link. The matrix for a given link is updated using a table of matrices of the correct group symmetry. The program permits the choice of a cold (ordered) or hot (disordered) start.

Restriction on the complexity of the program
In practice, the storage requirement is crucially connected with the array ALAT which stores the link matrices for a given configuration on the lattice. This array is placed via a LEVEL2 statement in the LARGE CORE MEMORY of the CDC 7600 computer, the statement being ignored by the CDC 6600 computer. ALAT is a complex array requiring a total storage of $2DS^DN^2$ words, where D is the dimensionality of the lattice

* The submitted manuscript was written under contract DE-AC02-76CH00016 with the US Department of Energy.

space, S the number of sites per dimension and N the degree of the group (i.e., $U(N)$ or SU(N)). For efficient runs N should be 2 or more. The $U(1)$ case is added to the program merely for completeness and for testing the program against other $U(1)$ programs. It is inefficient for two reasons:

(i) The heat bath method is usually more efficient than the Metropolis algorithm for $U(1)$.
(ii) The program for uniformity employs 1×1 arrays for the $U(1)$ case.

Clearly from a computer point of view the location of real or complex variables should prove more efficient than for real or complex 1×1 arrays. If a user wishes to make a series of $U(1)$ runs it would be better to use a program such as that in ref. [4] to produce the results. It should be noted that for $U(1)$ the fundamental and adjoint representations are identical. For the test run N, S, D took the values 2, 4, 4, respectively. Certain other arrays in the program, to be found in COMMON BLOCKS throughout the program, and also as local arrays in subroutines MONTE and RENORM, are dependent for their dimensions on the values of N and D. Comments in the program indicate how these arrays should be dimensioned.

Typical running time
The execution time increases with the number of links, the degree N of the group and the number of complete Monte Carlo iterations (or "passes") through the lattice. It is also dependent on the value for NTMAX ("number of hits per link") used. It increases with NTMAX though convergence towards equilibrium is accelerated. There can be an ultimate payoff in having NTMAX fairly large, say 20. For the test run NTMAX was set 5, S and D set 4, and the time for the 15 SU(2) iterations shown, was 109 s (i.e. ≈ 0.1 s per link) for the CDC 6600 computer, the CDC 7600 being approximately 5 times faster.

References

[1] K.G. Wilson, Phys. Rev. D10 (1974) 2455.
[2] A.M. Polyakov, Phys. Lett. 59B (1975) 82.
[3] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller and E. Teller, J. Chem. Phys. 21 (1953) 1087.
[4] R.C. Edgar, L. McCrossen and K.J.M. Moriarty, Comput. Phys. Commun. 22 (1981) 433.

LONG WRITE-UP

1. Introduction

It has been the dream of physicists to produce a grand unification scheme with which all the forces in nature could be described by a single theory – Grand Unification Theory (GUT). Maxwell in the nineteenth century successfully unified the theories of electricity and magnetism. Weinberg [1] and Salam [2] were able to bring electromagnetism and the weak interaction into a single theory, exploiting a Yang–Mills gauge theory based on an SU(2) ⊗ $U(1)$ group. Gauge theories have become established as the framework for building GUTs (see books like those listed in ref. [3]). Georgi and Glashow [4] have proposed an SU(5) model for unification of the theory of strong, weak and electromagnetic interactions with the breakdown into the respective groups for the interactions as:

$$SU(5) \rightarrow SU(3) \otimes SU(2) \otimes U(1).$$

Because of certain inadequacies or controversial features of the SU(5) model, even higher SU(N) groups have been proposed as candidates for GUTS:

SU(6): without proton decay [5];
SU(7): motivated by the persistently elusive nature of the t-quark in experiments and claims for observation of fractional charge $(+\frac{1}{3}e)$ [6];
SU(8): draw backs in SU(5) theory (see ref. [7]), which is otherwise quite successful, may be overcome by taking the fundamental particles that appear in SU(5) as a subset of the SU(8) bound state of preons that belong to SO(8) extended supergravity [8].

Chaichian, Kolmakov and Nelipa [9] have argued from a set of general requirements that only SU(6), SU(7), SU(8) form allowed groups out of the general SU(N) gauge group. In any case, large gauge groups are worthy of further study.

Quantum electrodynamics (QED) has proved very successful in describing the electromagnetic force. QCD (quantum chromodynamics) as a description of the strong force has resisted such detailed analysis, largely because of non-perturbative effects. Lattice gauge theories proposed by Wilson [10] provide a new calculational tool and have given the best evidence that QCD gives rise to quark confinement (i.e., that quarks cannot be isolated). As a preliminary to a study of the full theory coupled to fermion fields, considerable simplification can be achieved by using pure gauge groups on a lattice. Thus, $U(1)$ can provide a discrete version of electrodynamics, $U(2)$, an approximation to $SU(2) \otimes U(1)$ of the Weinberg–Salam electroweak theory, and SU(3) (or even SU(2)) for strong interactions.

Further simplifications can be made within the framework of the lattice itself. Periodic boundary conditions, employed within the program, enable the fields to be extended throughout space, though the choice of the number of sites per period needs to be taken with some care. It should be quite small to speed computation but not so small that spurious effects due to the periodicity are introduced.

Dimensional considerations provide another area of interest. Creutz [11] examined pure SU(2) gauge fields in four and five space–time dimensions. He found that five-dimensional SU(2) fields exhibited a phase transition whereas its four-dimensional counterpart did not, i.e., the confinement of quarks is connected to the dimensionality of space–time. Bhanot and Creutz have numerically confirmed the arguments of Polyakov [12] that for three space–time dimensions, $U(1)$ exhibits only a single phase whereas for four or five dimensions, deconfining phase transitions are present [12]. Balian, Drouffe and Itzykson have presented arguments that in a sufficient number of space–time dimensions, any lattice theory will exhibit such a phase transition [13]. Others have examined 1 + 1 or 2 + 1 space–time either because of similarities between the former and the full 3 + 1 space–time [14] or because certain theories can be examined on a lattice and the results compared with those derived using non-Monte Carlo methods (e.g., variational or even analytic) [15]. The hope in the latter case is that some features of the theory (e.g., QCD) will thereby emerge.

The program is designed to allow varying numbers of dimensions to be examined with a flexibility in numbers of lattice sites, for pure SU(N) and $U(N)$ gauge groups.

Considerable interest has also centred around the choice of action. The form of the action to be used on the lattice is not unique, in contrast to the desired continuum limit. The traditional and simplest form is that adopted by Wilson [10]. Other forms have been used by Manton [16], Villain [17] and others. Another approach is to take the adjoint representation which for SU(N) and $U(N)$ is equivalent to factoring out the center of the group. This has been employed by a number of research groups [18]. The adjoint trace Tr_A is connected to the fundamental trace Tr_F by the simple relationship

$$\mathrm{Tr}_A(U) = |\mathrm{Tr}_F(U)|^2 - 1$$

for SU(N) and, removing a trivial singlet piece, for $U(N)$ also. Thus, since $\mathrm{Tr}_F(U)$ represents the Wilson action per plaquette where U is the product of the group matrices around the plaquette, the action in the adjoint representation can be easily calculated. The reason for the interest shown in alternative forms of the action lies in the presence of new phase transitions absent with the Wilson action. This is true for SU(2) and SU(3); see, for example, ref. [18]. Of course, there is no guarantee that the phase transitions produced are deconfining. The program enables phase diagrams to be produced using different values of the coupling constants for the fundamental and adjoint representations, as shown in the papers of Bhanot and Creutz, and Bhanot [18].

The Monte Carlo simulation bears a close correspondence to statistical mechanics and algorithms for generating statistical equilibrium can be employed. Two of the most popular are the "heat bath" method of Yang [19] introduced to gauge theory by Creutz [20], and the Metropolis algorithm [21]. The "heat bath" algorithm has the advantage in that for continuous groups with simple manifolds, such as $U(1)$ and SU(2), it can result in savings of computer time over the Metropolis algorithm [20]. However, it suffers from the disadvantage that it requires detailed knowledge of the Haar measure for each group used. Pietarinen has used the heat bath approach in examining the string tension in SU(3) lattice gauge theory [22], in order to enable calculations to be performed on large lattices. We have used the

Metropolis algorithm here, although in previous published programs for $U(1)$ [23] and SU(2) [24], the heat bath method was adopted.

2. Code description

The program test deck used to produce the test output consists of five routines SUUNFA, WRTOUT, MONTE, RENORM and STATS.

(i) SUUNFA is the program routine. It contains general comments on the program, on the library routines needed, on the input data required and on how to set up the COMMON blocks. It also reads in the data, all of which is set up in the form of floating point numbers. The data is written out again for convenience using a call to WRTOUT. Various parameters are initialized. The link matrices, stored in ALAT, are all set to identity matrices. This puts the program to the configuration of an ordered ("cold") start. If the input data requires a disordered ("hot") start, then one complete pass is made through the lattice with the inverse temperature β_F (B, in program) set small (0.00001) and the other inverse temperature β_A (BA, in program) set to zero. This switches off the adjoint part of the action and the small value of β_F causes ALAT to be replaced by link matrices whose elements are reasonably randomly distributed. The parameters β_F and β_A are then reset to their input values. Calls to MONTE and RENORM are made. RENORM is only called approximately every 50 passes through the lattice, as this usually provides enough correction to ALAT for loss of its SU(N) or $U(N)$ symmetry through machine rounding errors.

(ii) WRTOUT causes the input data to be written out for reference.

(iii) MONTE makes a specified number of complete sweeps (called "iterations" in the program) through the lattice. It first sets up a table of random matrices of the correct SU(N) or $U(N)$ symmetry. This is done for each sweep through the lattice. These provide matrices for transforming the link matrices to new values to be accepted or rejected with the Metropolis algorithm. For a given link several such attempts (NTMAX in number) are made before MONTE passes on to the next link. This can result in considerable saving of computing time. The value of NTMAX may be chosen by making trial runs. Values between about 5 and 20 give reasonable performance in most cases.

Two features in the construction of the table of matrices are included to help improve efficiency [25]. They are under the control of the user, and are: (a) BEFF. This controls the bias of these random matrices towards the identity. It is designed to become more important for large β_F or β_A, when there should be a bias towards the completely ordered lattice configuration (when β_F or $\beta_A \to \infty$). The actual form of BEFF can be varied by the user to improve convergence to equilibrium. Symmetrization to SU(N) or $U(N)$ form of the table of matrices is then carried out. (b) ZN. A fraction of the matrices of the table were biassed towards the center Z_N of the SU(N) or $U(N)$ group. This again is chosen to aid convergence, and is most important when β_A is large.

To include matrices which bias in the opposite direction, only half the table is set up, the remainder of the matrices being their inverses which are readily generated simply by taking the transpose and complex conjugate of each matrix, due to the unitarity.

(iv) RENORM resymmetrizes the link matrices stored in the matrix ALAT. This is necessary to overcome the effect of rounding errors produced by the computer. It will be dependent on the word length used by the computer. For the CDC 6600 or 7600 computers with a 60 bit word length (giving an accuracy of about 14 significant figures) usually a RENORM call was only necessary every 50 complete sweeps through the lattice.

(v) STATS produces calculations of the mean and standard deviations of the results as each sweep of the lattice is produced. This is achieved by storing the sum over the number of sweeps of the various quantities and their squares. Division of the sum by the number of sweeps considered gives the mean and the square root of the difference between the sum of the squares divided by the number of sweeps and square of the mean gives the standard deviation. Monte Carlo calculations have a relaxation time and a certain number of sweeps are necessary to produce equilibrium. To eliminate these "relaxation" sweeps, the first

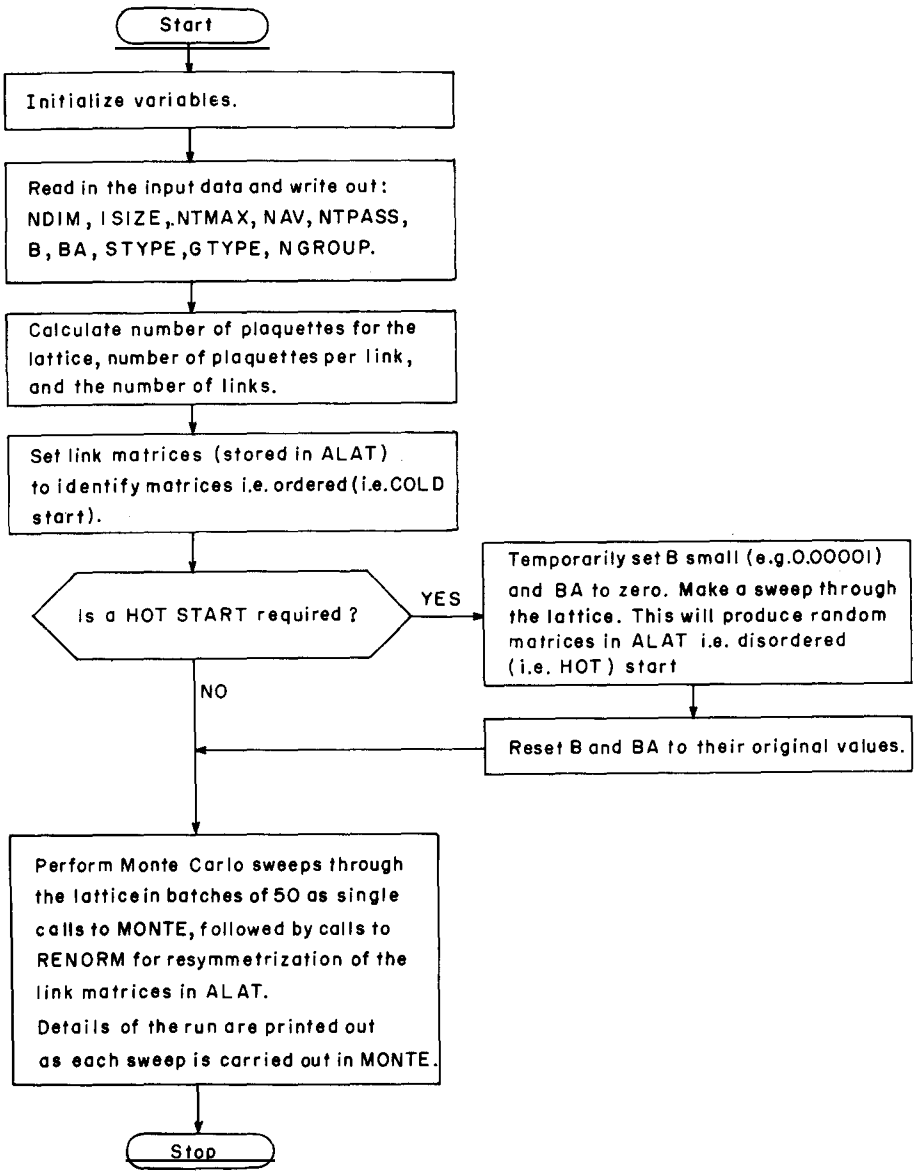
Start
Initialize variables.
Read in the input data and write out: NDIM, ISIZE,.NTMAX, NAV, NTPASS, B, BA, STYPE, GTYPE, NGROUP.
Calculate number of plaquettes for the lattice, number of plaquettes per link, and the number of links.
Set link matrices (stored in ALAT) to identify matrices i.e. ordered (i.e. COLD start).
Is a HOT START required ?
YES
NO
Temporarily set B small (e.g. 0.00001) and BA to zero. Make a sweep through the lattice. This will produce random matrices in ALAT i.e. disordered (i.e. HOT) start
Reset B and BA to their original values.
Perform Monte Carlo sweeps through the lattice in batches of 50 as single calls to MONTE, followed by calls to RENORM for resymmetrization of the link matrices in ALAT.
Details of the run are printed out as each sweep is carried out in MONTE.
Stop

Fig. 1. Flow chart of the program.

NAV (parameter set in input data) sweeps can be ignored in the calculation of the mean and standard deviation. To facilitate the printout of the results, fatuous use is made of a CDC feature: if the number is too large to fit the format, a series of *'s is output. Thus by setting the means and standard deviations to arbitrary large numbers for the first NAV sweeps, the means and standard deviations will be written as a series of * for these sweeps. The user of another make of computer may have to alter this technique or the machine will register an error.

Caution is necessary for the interpretation of these thermal fluctuations as statistical errors because successive iterations are highly correlated. A simple way to estimate the true errors on a long run is to divide it into a few blocks, each long enough to be uncorrelated with the others. A standard deviation of the mean may then be calculated from the averages over the separate blocks.

The flow chart of the program is shown in fig. 1.

3. Specification of the lattice

MONTE is the most important routine in the program. The other routines, apart from RENORM, provide input to MONTE, output from it or analysis of its results. The flow chart for MONTE is presented in fig. 2.

The lattice is accessed in an orderly fashion. It is traversed in a similar manner to that described in ref. [24], but some slight differences. The theory, described in section 5, requires evaluation of products of link matrices taken around plaquettes (elementary squares lying on the coordinate planes and whose corners are adjacent sites). For efficiency of calculation, the plaquettes are grouped, in pairs lying in the planes, around common links. This is illustrated for a 3 dimensional lattice in fig. 3. The six sites of a pair of plaquettes are labelled by numbers enclosed in circles, and the center link runs between ① and ⑤.

To impose the periodic boundary condition in a simple and efficient way, we introduce the arrays MUP and MDOWN. Modulo ISIZE, these arrays give their index shifted up or down, respectively, by one unit.

To progress through the lattice the following steps are carried out:

i) Select a location for the key site ① of the plaquette group. This is done in sequence in a manner which causes coordinates to change faster from the left (e.g., for a lattice with 3 sites per coordinate direction, i.e., ISIZE = 3, we have the sequence: (1, 1), (2, 1), (3, 1), (1, 2), (2, 2)..., in two dimensions (i.e., NDIM = 2) or (1, 1, 1), (2, 1, 1), (3, 1, 1), (1, 2, 1), (2, 2, 1),..., in three dimensions (i.e., NDIM = 3). In the program the values of the coordinates of ① are stored in the integer array X with X(1) the first coordinate value, X(2) the second and so on.

ii) Next for a certain choice of site ① we select a link from ① to ⑤. This is done in sequence according to values of a variable I1 running from 1 to NDIM. I1 = 1: ⑤ will have the same coordinates as ① except that the first coordinate is increased by 1, I1 = 2: ⑤'s second coordinate, instead, increased by 1, over the respective coordinate of ①, and so on.

iii) For this choice of link, (NDIM-1) double plaquettes lying in planes are constructed with this link as common center link. These planes are labelled by values of a variable I2 running from 1 to NDIM but not including the current value of I1. I2 can be thought of as controlling the value of the site labelled ② in a manner similar to that of I1 on ⑤, i.e., for: I2 = 1: ② will have its first coordinate only differing by one unit from the coordinates of ① and so on.

For a given common link, the individual plaquettes are numbered (by variables IPL1, IPL2 and IP in the program). This is needed in calculations in the adjoint representation.

Fig. 4 shows how the doubled plaquettes are oriented for different values of the direction variable I1 for a 2 dimensional lattice, appropriate to selections of site ① from the lattice shown in fig. 5. Fig. 5 shows a two-dimensional lattice with links drawn as unbroken lines. In order to illustrate the periodic continuations, extra links are drawn as dashed lines. The two dotted lines are added for aesthetic reasons only. The crucial parameters that are needed for the calculation are the numbers assigned to the links (including the continuation links). These enable the link matrices appropriate to each link to be extracted from and re-stored in the matrix ALAT. In the program the value of L12 gives the link parameter for the link from site ① to site ②, L26 that from ② to ⑥ and so on. The link parameters are shown in square brackets in fig. 5. As a preliminary to obtaining these L parameters, the program calculates numbers assigned to the sites of the lattice. For a given double plaquette, M1 gives the number for site ①, M2 for ② and so on. The full set of site numbers is shown without brackets on fig. 5.

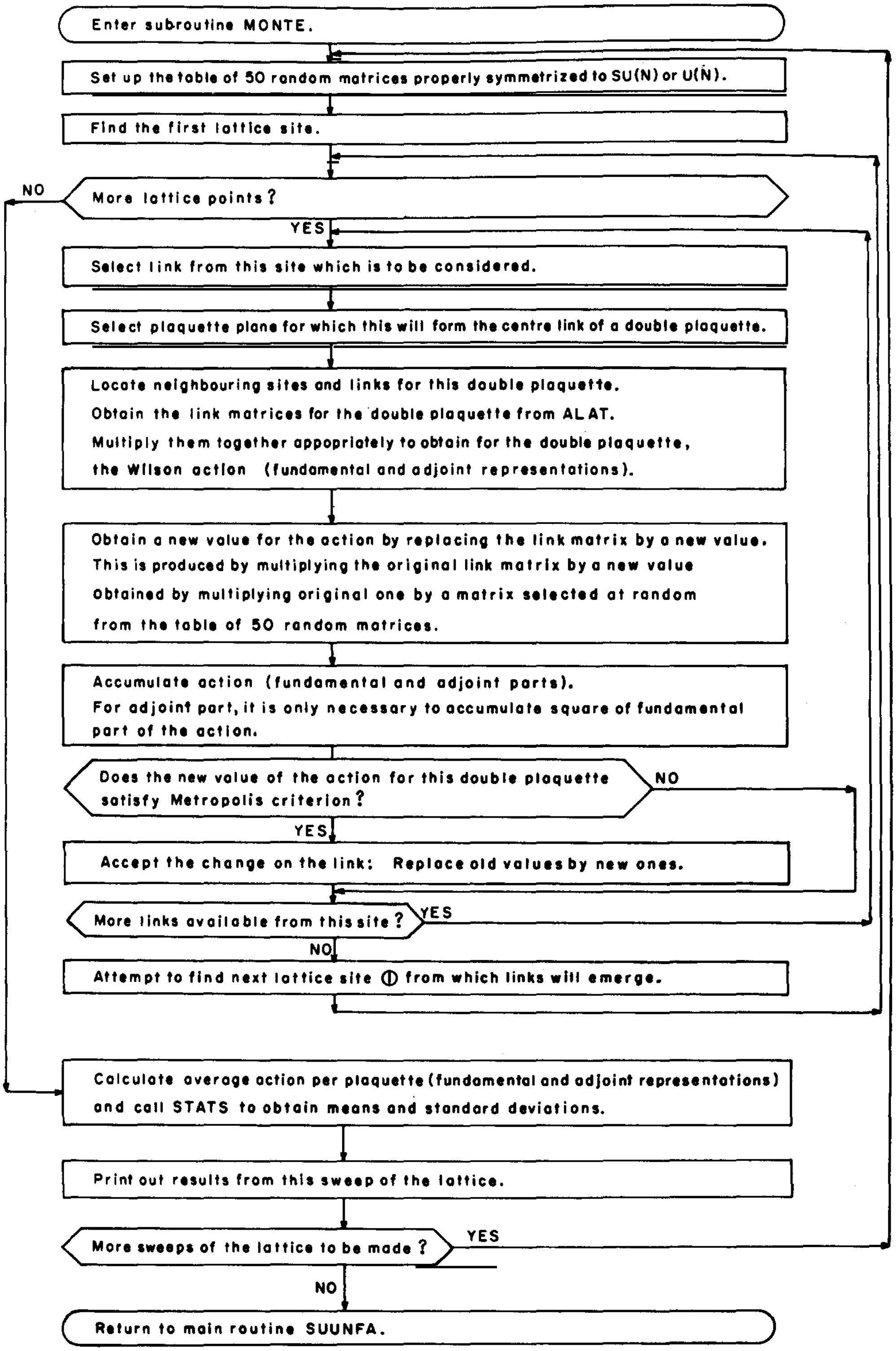
Enter subroutine MONTE.
Set up the table of 50 random matrices properly symmetrized to SU(N) or U(N).
Find the first lattice site.
NO
More lattice points ?
YES
Select link from this site which is to be considered.
Select plaquette plane for which this will form the centre link of a double plaquette.
Locate neighbouring sites and links for this double plaquette.
Obtain the link matrices for the double plaquette from ALAT.
Multiply them together appopriately to obtain for the double plaquette,
the Wilson action (fundamental and adjoint representations).
Obtain a new value for the action by replacing the link matrix by a new value.
This is produced by multiplying the original link matrix by a new value
obtained by multiplying original one by a matrix selected at random
from the table of 50 random matrices.
Accumulate action (fundamental and adjoint parts).
For adjoint part, it is only necessary to accumulate square of fundamental
part of the action.
Does the new value of the action for this double plaquette
satisfy Metropolis criterion ?
NO
YES
Accept the change on the link: Replace old values by new ones.
More links available from this site ?
YES
NO
Attempt to find next lattice site ① from which links will emerge.
Calculate average action per plaquette (fundamental and adjoint representations)
and call STATS to obtain means and standard deviations.
Print out results from this sweep of the lattice.
More sweeps of the lattice to be made ?
YES
NO
Return to main routine SUUNFA.

Fig. 2. Flow chart of subroutine MONTE.

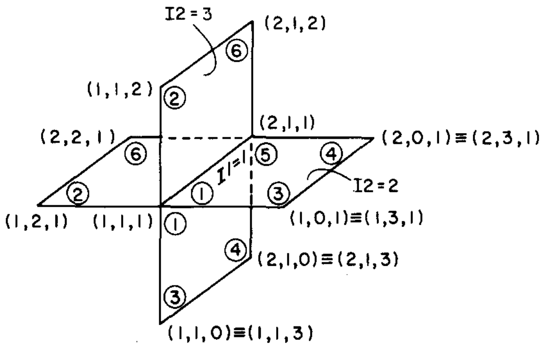

Fig. 3. Two double plaquettes with a common center link (between sites ① and ⑤) from a 3-dimensional lattice. The variables I1 and I2 are the loop variables appearing in subroutine MONTE. I1 labels the common center links emanating from a particular site chosen for ① and I2 the double plaquettes with this as common center link.

4. The link matrices

The matrix ALAT is the dominant user of computer memory, especially if the degree N of the groups SU(N) or $U(N)$ used is large and the number of lattice sites is also large. Thus ALAT is placed in LEVEL 2 – the large core memory (LCM) on CDC 7600. This is not quite so quickly accessed as the small core memory (SCM), but is usually larger and cheaper to use. LEVEL 2 statements are ignored on the CDC 6600, which does not have the feature of distinguishing between LCM and SCM. For other computers the LEVEL 2 statements will probably not be recognized and should be removed.

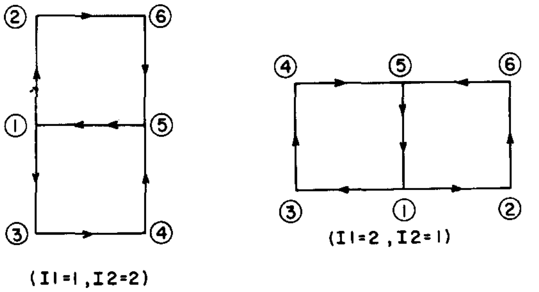

Fig. 4. Two double plaquettes with common links orientated in the two directions possible for a 2-dimensional lattice. These are representative of those present in the 2-dimensional lattice given in fig. 5. The site numbering and parameters I1 and I2 are as for fig. 3.

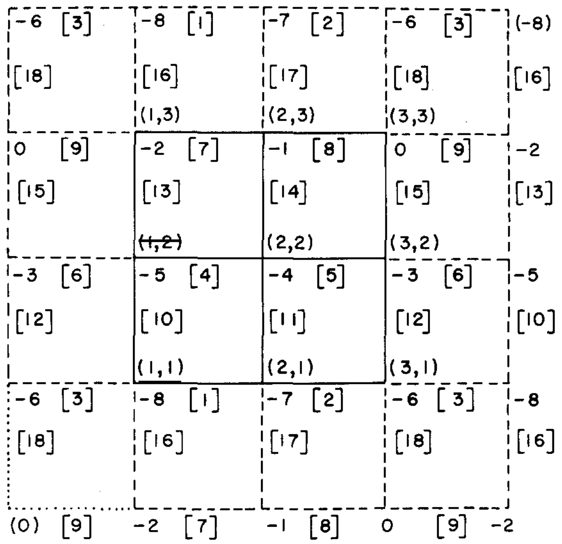

Fig. 5. Link and site numbers and coordinates of sites for a 2-dimensional lattice with 3 sites per dimensional direction. Link numbers have square brackets, site numbers have no brackets and coordinates round brackets. The single numbers in brackets are site numbers added for completeness. The dotted lines are extensions to the lattice to produce periodic links across the boundary. The dotted lines give a plaquette added for completeness.

The first index of ALAT labels the corresponding link of the lattice. It must be dimensioned at least to NDIM × ISIZE$^{\mathrm{NDIM}}$, where ISIZE is the number of sites per dimension and NDIM the number of dimensions. The other two indices have both dimension N for the groups SU(N) or $U(N)$ on the lattice. They are the matrix indices for the group elements.

The link matrices are constructed as follows. They are set originally to identity matrices I_N in the main routine SUUNFA. They are then modified in the routine MONTE by multiplying by matrices, of the correct SU(N) or $U(N)$ symmetry, under the control of the Metropolis algorithm [21]. The latter matrices are generated in MONTE and stored in a matrix A. Some comments on the contruction of the matrices in A are given above in section 2. The production of the table of matrices, i.e., A, is carried out as follows:

i) Set the elements of half of the matrices (i.e., 25 in number) of A to complex numbers whose real and imaginary parts are random numbers, uniformly distributed, lying between $-1/2$ and $1/2$, except for the real parts of the diagonal elements, which lie between $-1/2$ and BEFF $-1/2$.

ii) Each matrix of the table is then reduced to SU(N) or $U(N)$ symmetry; this is done by successively working through it row by row, first normalizing the current row and then projecting it out of the subsequent rows of the matrix; on the completion of this process the matrix will be unitary – indeed, this is the Gram–Schmidt orthogonalization process [26]; the matrix has now $U(N)$ symmetry;

iii) If SU(N) symmetry is required, the matrix after stage (ii) is reduced to unit determinant;

iv) The first 5 of the 25 matrices are multiplied by $\exp(2\pi i/N)$; this does not alter the SU(N) or $U(N)$ symmetry;

v) using the unitary nature of matrices, their inverses are generated and become the remainder of the table of matrices.

The stages (ii) and (iii) given above ((iii) only used if required) form the basis of the routine RENORM which serves to resymmetrize ALAT, every 50 (or any other appropriate number, so determined by the user) sweeps through the lattice. The loss of symmetry arises due to the rounding errors of the computer. Obviously the smaller the precision of numbers capable of being stored in the computer (due to small word length used in the computer) and the values making up $\langle E \rangle$, the more marked will be this effect.

It should be noted that the matrices assigned in ALAT to the links are directed, i.e., if the link is taken from one site to another such that any coordinate between the sites is increased, then the matrix on the link is taken without alteration, otherwise the inverse is used. The unitary nature of the matrices will facilitate the acquisition of the inverse. The need for directed links will become clear in the next section.

5. Underlying theory of the algorithms used

Wilson's lattice gauge theory [10] is based on evaluation of link matrices around plaquettes, with appropriate link directions, as described in the last section. Thus, in evaluating the matrix product $U_\square$ on a plaquette, the product of the four link matrices is involved. For a pair of plaquettes showing a link, the directions are shown in fig. 4 by arrows. The natural direction of the links has to be changed only in the links from ⑥ to ⑤ from ① to ③ and ① to ⑤. This means that the link matrices taken from ALAT in these cases have to be inverted. For a group of plaquettes with a common link, only the link matrix on this common link is updated in the Metropolis algorithm. This makes the NTMAX Metropolis applications (described in section 2) an efficient process as the bulk of the calculation for $U_\square$ for the plaquette cluster need not be repeated.

The action, following the theory of Wilson, is written

$$S = \sum_{\square} S_\square, \tag{1}$$

where the sum extends over all plaquettes □, and $S_\square$ is a function only of $U_\square$. Following Bhanot and Creutz [25], we generalize their equation (3) to

$$S_\square = \beta_{\mathrm{F}}\left[1 - \frac{1}{N}\,\mathrm{Re}\,\mathrm{Tr}\,U_\square\right] + \beta_{\mathrm{A}}\left[1 - \frac{1}{(N^2-1)}\,\mathrm{Tr}_{\mathrm{A}} U_\square\right]. \tag{2}$$

This formula also appears in other papers (c.f., for example ref. [27]). The program uses the relation

$$\mathrm{Tr}_{\mathrm{A}} U = |\mathrm{Tr}\,U|^2 - 1, \tag{3}$$

connecting the trace of the matrix U in the fundamental representation with the trace in the adjoint representation. (For $U(N)$, the adjoint representation contains a singlet piece which the -1 removes.) Eq. (3) means that the calculation of $S_\square$ for only the fundamental part can be readily used to get the full result of (2). In the program, this means that both the sum of Tr $U_\square$ and the sum of $|\mathrm{Tr}\,U_\square|^2$ have to be accumulated for all the plaquettes with a common link.

There is a strong correspondence between lattice gauge theory and statistical mechanics. This is reflected in the path integral

$$Z = \int \mathrm{e}^{-S}\mathrm{d}U \tag{4}$$

which can be regarded as a partition function. In (4) dU is the invariant group measure or Haar measure. This is the great advantage of using the Metropolis algorithm, as it avoids needing detailed knowledge of the Haar measure.

Motivated by the two parts of the action, we consider the average over the lattice of the quantities appearing in square brackets in (2). These are the generalization of the quantities of eqs. (9) and (10) of ref. [25]:

$$P = \langle 1 - \frac{1}{N} \operatorname{Re} \operatorname{Tr} U_{\square} \rangle, \tag{5}$$

$$P_{\mathrm{A}} = \langle 1 - \frac{1}{(N^2-1)} \operatorname{Tr}_{\mathrm{A}} U_{\square} \rangle. \tag{6}$$

These are called, respectively, fundamental plaquette and adjoint plaquette in the test run output of the program. For an action with only one parameter, e.g., $S_{\square}$ of (2) with either β_{F} or β_{A} zero, the corresponding quantity of eqs. (5) and (6) is just the average energy $\langle E \rangle$ per plaquette.

The Metropolis algorithm [21] can be easily adapted to evaluate these expectations. Following [21] and adapting to the present case, the average of a variable F is

$$\langle F \rangle = \left[\int F \, \mathrm{e}^{-S} \mathrm{d}U \right] / Z. \tag{7}$$

Here F can be identified with either of the square brackets of (2) and $\langle F \rangle$ with P or P_{A}, as appropriate. Suppose we take a new configuration of the system obtained by changing one link matrix (by multiplying it by one of the matrices of the table of matrices). If the change in S, $\Delta S < 0$, i.e., new configuration lowers S, then this new configuration is accepted (i.e., lattice matrix ALAT is altered to this configuration). If $\Delta S > 0$ we accept the new configuration with probability $\mathrm{e}^{-\Delta S}$; this means taking a random number x between 0 and 1, and accepting the configuration if $x < \mathrm{e}^{-\Delta S}$, otherwise remaining at the old configuration. We then sum up all the values of F for all plaquettes involving the given link whether we have gone to a new configuration or stayed in the old one. We divide by the total effective number of plaquettes counted to get $\langle F \rangle$ of (7).

We also include in the test run output the sweep by sweep value of $\langle S \rangle$ per plaquette, i.e., action per plaquette.

6. Adaptations

The program can be readily adapted to evaluate Wilson loops (and hence the string tension) and plaquette–plaquette interactions (and hence mass gaps).

References

[1] S. Weinberg, Phys. Rev. Lett. 19 (1967) 1264.

[2] A. Salam, in: Relativistic Groups and Analyticity, ed. N. Svartholm (Interscience, New York, 1968).

[3] C. Itzykson and J.-B. Zuber, Quantum Field Theory (McGraw Hill, New York, 1980).
L.D. Faddeev and A.A. Slavnov, Gauge Fields: Introduction to Quantum Theory (Benjamin/Cummings, New York, 1980).
I.J.R. Aitchison and A.J.G. Hey, Gauge Theories in Particle Physics (Hilger, London, 1982).

[4] H. Georgi and S.L. Glashow, Phys. Rev. Lett. 32 (1974) 438.

[5] M. Singer and K.S. Viswanathan, A Grand Unified Tumbling Gauge Theory based on SU(6) (Simon Fraser University, Burnaby, May 1981).

[6] J.E. Kim, Phys. Rev. Lett. 45 (1980) 1916; Phys. Rev. D23 (1981) 2706.
P. Cox, P. Frampton and A. Yildiz, Phys. Rev. Lett. 46 (1981) 1051.
E. Farhi and L. Susskind, Phys. Rev. D20 (1979) 3404.
I. Umemura and K. Yamamoto, Phys. Lett. 100B (1981) 34.
Z. Ma, T. Tu, P. Xue and Z. Yue, Chinese Institute of High Energy Report (1980).
M. Claudson, A. Yildiz and P.H. Cox, Phys. Lett. 97B (1980) 224.
K. Kang and I.-G. Koh, Fermilab Publ. 81/70-THY (October 1981).

[7] G. Lazarides and Q. Shafi, Phys. Lett. 99B (1981) 113.

[8] J. Ellis, M.K. Gaillard and B. Zuenino, Phys. Lett. 94B (1980) 343.
J. Ellis, M.K. Gaillard, L. Maiani and B. Zumino, LAPP preprint TH15/CERN preprint TH 2481 (1980).
E. Cremmer and B. Julia, Phys. Lett. 80B (1978) 48; Nucl. Phys. B159 (1979) 141.

[9] M. Chaichain, Yu.N. Kolmakov and N.F. Nelipa, Nucl. Phys. B186 (1981) 257.

[10] K. Wilson, Phys. Rev. D10 (1974) 2455.

[11] M. Creutz, Phys. Rev. Lett. 43 (1979) 553.

[12] G. Bhanot and M. Creutz, Phys. Rev. D21 (1980) 2892.
A.M. Polyakov, Nucl. Phys. B120 (1977) 429.

[13] R. Balian, J.M. Drouffe and C. Itzykson, Phys. Rev. D10 (1974) 3376, D11 (1975) 2098, 2104.

[14] A.A. Migdal, Zh. Eksp. Teor. Fiz. 69 (1975) 810 (Sov. Phys. JETP 42 (1976) 413).
L.P. Kadanoff, Rev. Mod. Phys. 49 (1977) 267.
J.B. Kogut, Rev. Mod. Phys. 51 (1979) 659.
G. Immirzi, Preprint HUTMP 80/B102, Lyman Lab. of Phys. Harvard University (1980).
J. Shigemitsu, J.B. Kogut and D.K. Sinclair, Preprint I11-(TH)-80-52, University of Illinois at Urbana-Champaign (December 1980).
G. Fox, R. Gupta, O. Martin and S. Otto, Preprint CALT-68-866, California Institute of Technology (November 1981).
J. Kogut, M. Snow and M. Stone, Phys. Rev. Lett. 47 (1981) 1767.

[15] J. Greensite, Nucl. Phys. B166 (1980) 113.
A. Patkös and F. Deak, Z. Phys. C9 (1981) 359.
A. Patkös, Preprint NBI-HE-81-41, Niels Bohr Institute (October 1981).
R.P. Feynman, Nucl. Phys. B188 (1981) 479; A Qualitative Discussion of Quantum Chromodynamics in 2+1 Dimensions, Lisbon (9–15 July 1981).

[16] N.S. Manton, Phys. Lett. 96B (1980) 328.

[17] J. Villain, J. de Phys. 36 (1975) 581.

[18] I.G. Halliday and A. Schwimmer, Phys. Lett. 101B (1981) 327.
J. Greensite and B. Lautrup, Phys. Rev. Lett. 47 (1981) 9.
G. Bhanot and M. Creutz, Phys. Rev. D24 (1981) 3212.
G. Bhanot, IAS preprint (September 1981).
Yu.M. Makeenko and M.I. Polikarpov, Preprint ITEP-152 (1981).
M. Creutz and K.J.M. Moriarty, Preprint BNL-31037 (February 1982).

[19] C.-P. Yang, Proc. Symp. in Applied Mathematics, Vol. XV, Am. Math. Soc., Providence, RI (1963).

[20] M. Creutz, Phys. Rev. D21 (1980) 2308.
See also: M. Creutz, L. Jacobs and C. Rebbi, Phys. Rev. D20 (1979) 1915.

[21] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller and E. Teller, J. Chem. Phys. 21 (1953) 1087.

[22] E. Pietarinen, Helsinki preprint HU-TFT-80-49 (1980).

[23] R.C. Edgar, L. McCrossen and K.J.M. Moriarty, Comput. Phys. Commun. 22 (1981) 433.

[24] R.W.B. Ardill and K.J.M. Moriarty, Comput. Phys. Commun. 24 (1981) 989.

[25] G. Bhanot and M. Creutz, Phys. Rev. D24 (1981) 3212.

[26] H.S.W. Massey and H. Kestleman, Ancilliary Mathematics (Pitman, London, 1964) p. 982.

[27] M. Creutz and K.J.M. Moriarty, Preprint BNL-31037 (February 1982).

TEST RUN OUTPUT

```
********************************************************************************
*                                                                              *
*  LATTICE GAUGE THEORY : FUNDAMENTAL AND ADJOINT REPRESENTATIONS              *
*  VERSION : SUUNFA244 CY=7 29/6/1982 NGROUP=2,ISIZE=4,NDIM=4,NTMAX=5          *
*  JOB SUBMITTED : 29/6/1982 (J9,T120).                                        *
*                                                                              *
********************************************************************************


GROUP = SU( 2)
        ------
LATTICE WITH  4 DIMENSIONS AND  4 SITES PER DIMENSION
              --                --
BETA (FUNDAMENTAL) =    .550 AND BETA (ADJOINT) =   2.340 WITH A COLD START
                     -------                      -------        ----
NUMBER OF PASSES TAKEN ON A LINK (NTMAX) IS   5
                                             ---
MEAN AND STANDARD DEVIATION TAKEN IGNORING THE FIRST     5 PASSES
                                                     -----
TOTAL NUMBER OF PASSES IS    15
                          -----

PASS NUMBER         FUNDAMENTAL PLAQUETTE                    ADJOINT PLAQUETTE                      TOTAL ACTION
                 VALUE       MEAN       S. DEV.          VALUE       MEAN       S. DEV.          VALUE       MEAN       S. DEV.


         1       .022154   *********   *********        .057305   *********   *********        .146278   *********   *********
         2       .056855   *********   *********        .133272   *********   *********        .343126   *********   *********
         3       .090877   *********   *********        .198349   *********   *********        .514118   *********   *********
         4       .121224   *********   *********        .251789   *********   *********        .655859   *********   *********
         5       .130153   *********   *********        .280828   *********   *********        .728722   *********   *********
         6       .149631     .149631    0.000000        .298289     .298289    0.000000        .780292     .780292    0.000000
         7       .159151     .154391     .004760        .305842     .302065     .003777        .803203     .791748     .011456
         8       .158059     .155614     .004254        .310682     .304937     .005100        .813928     .799141     .014029
         9       .164349     .157798     .005280        .312250     .306766     .005434        .821056     .804620     .015416
        10       .180703     .162379     .010308        .319523     .309317     .007047        .847071     .813110     .021874
        11       .199405     .168550     .016702        .340377     .314494     .013243        .906154     .828617     .040014
        12       .193996     .172185     .017844        .355230     .320313     .018802        .937936     .844234     .053251
        13       .181269     .173320     .016959        .348381     .323822     .019887        .914910     .853069     .055023
        14       .181387     .174217     .016189        .336199     .325197     .019149        .886469     .856780     .052928
        15       .186355     .175431     .015784        .337087     .326386     .018513        .891278     .860230     .051267
```