MONTECARLO TECHNIQUES IN THE MANY BODY PROBLEM

R. Guardiola
Departamento de Física Nuclear
Universidad de Granada
18071 GRANADA (Spain)

ABSTRACT

These lecture notes have been prepared with the objective of being an introductory course to Montecarlo methods in quantum problems, and particularly in the quantum many body problem. In this sense, it has been assumed that the reader has no previous knowledge on Montecarlo methods in general, and even on mathematical statistics. A particular emphasis has been put in the practical character of the notes, and many examples have been included (even with a working Fortran code) to stimulate the reader to practice by himself most of the algorithms introduced.

The course starts with a brief introduction to the statistical concepts of interest for the remaining chapters, and with a quite detailed description of the ways of generating random numbers. The powerful Metropolis algorithm is presented with some theoretical background in the third chapter. Physics starts in chapter four, which is devoted to Variational Montecarlo. Cahpters six and seven are devoted to Diffusion Montecarlo and to Green function Montecarlo, respectively. These two theories share many ideas, and should not be read independently.

Along the lecture notes, some physical systems have been briefly described, with a particular emphasis on helium atoms systems. This description is very incomplete, and the reader should consult the specialized works or reviews referred in the notes to complement the description.

The notes include (quite unusually) several Fortran programs. Most of them do not require big computers or long time runs, and even with a personal computer, can be satisfactorily run. These codes are far from being optimized, the major emphasis being in having a direct correspondence with the algorithms previously described. I suggest the reader to copy and run these programs, and realize that Montecarlo really works. This is a good stimulus to go forward with the subject.

Lectures delivered at the Asociación pro Centro Internacional de Física (Bogotá, Colombia), July 1986. To be published at the ACIF series (World Scientific, Singapore), S. Rosati, ed.

1. BASIC CONCEPTS OF STATISTICS

1.1 Random variables and probability distribution functions

Consider a physical event or an action, such as throwing a dice or shooting on a target, in such a manner that we may repeat it as many times as needed. The result of this action may be expressed in terms of a numerical value, by assigning a numerical value to the outcome of the action. So we associate a random variable x which takes a concrete value at each trial.

The exhaustive set of all possible results is the domain of the random variable, e.g. $(x_1, x_2, \dots, x_n)$. After many empirical trials we may observe that some of the outcomes appear more frequently than some others. To each of the outcomes we associate a probability, and the ordered set of all these probabilities (ordered in correspondence with the set of possible results), $(p_1, p_2, \dots, p_n)$ is the probability distribution of the random variable. The exhaustive character of the domain implies that the sum of the probabilities is 1.

Summarizing, a random variable is defined by means of its domain, the set of possible values, and its probability distribution, the set of corresponding probabilities.

A simple example is the random variable defined as the sum of points after throwing two dices. As we know very well we have:

Domain:	2	3	4	5	6	7	8	9	10	11	12
Prob.Distr:	$\frac{1}{36}$	$\frac{2}{36}$	$\frac{3}{36}$	$\frac{4}{36}$	$\frac{5}{36}$	$\frac{6}{36}$	$\frac{5}{36}$	$\frac{4}{36}$	$\frac{3}{36}$	$\frac{2}{36}$	$\frac{1}{36}$.

The random variable can be univariate, as in the previous example, or multivariate. We may also consider random variables which take values in the continuum, in which case the density distribution function is defined.

TABLE 1.1. Concepts in discrete and continuous random variables

	Density Distr. Function	Cumulative Distr. Function	Domain
Discrete random v.	$p_i = \mathrm{Prob}(X = x_i)$	$P_i = \mathrm{Prob}(X \leq x_i)$	$x_1, x_2, \dots, x_n$
Continuous random v.	$f(x)dx = \mathrm{Prob}(x \leq X \leq x+dx)$	$F(x) = \mathrm{Prob}(X \leq x)$	(a,b)

These concepts are summarized in Table 1. There are some simple properties that probability distribution functions must satisfy:

i) Must be positive definite,

$0 \leq p_i \leq 1$, or

$f(x) \geqslant 0$, but without an upper bound.

ii) Bounds on the cumulative distribution functions,

$0 \leqslant P_i \leqslant 1$, or

$0 \leqslant F(x) \leqslant 1$

iii) Both P and F are non-decreasing functions,

$P_i \geqslant P_j$, if $i \geqslant j$, or

$F(x_1) \geqslant F(x_2)$, if $x_1 \geqslant x_2$

iv) Normalization,

$P_n = 1$, or

$F(b) = 1$, where b is the upper limit of the continuous domain.

1.2 Examples of probability distributions

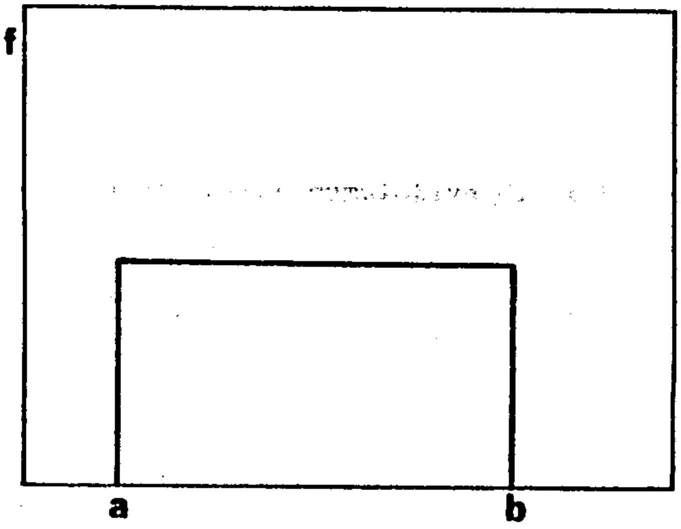


Figure 1.1. The uniform distribution

Uniform distribution U(a,b)

$f(x)=0, \; x < a$

$f(x)=1/(b-a), \; a \leqslant x \leqslant b \quad (1.1)$

$f(x)=0, \; x > b$

The standard form of this distribution corresponds to a=0 and b=1.This distribution is basic in the random number generation algorithms.

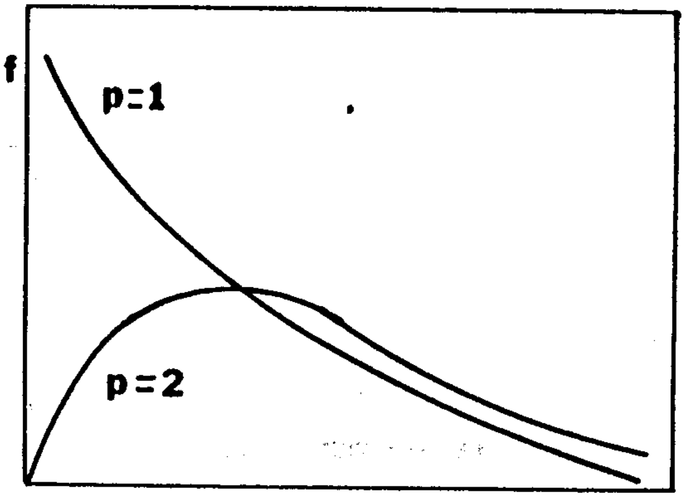


Figure 1.2 . The gamma distribution

Gamma distribution

$f(x)=x^{p-1}\exp(-x)/\Gamma(p) \quad (1.2)$

$0 \leqslant x < \infty$, and

$0 < p < \infty$.

This distribution is related to the Green function of a free particle in multi-dimensional spaces.

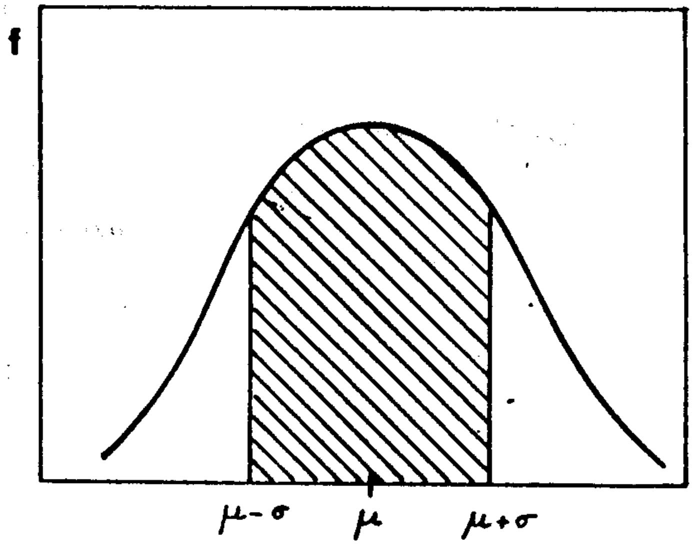


Figure 1.3. The normal distribution

Normal distribution

$$f(x)=\frac{1}{\sqrt{2\pi}}\,\frac{1}{\sigma}\,\exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\} \qquad (1.3)$$

μ and σ are parameters of the distribution, and

$$-\infty < x < \infty \ .$$

The normal distribution has been written so that its mean is μ_2 , and its variance is σ^2 . These concepts will be introduced below. The standard form corresponds to a mean of 0 and to a standard deviation $\sigma = 1$.

The value of σ is usually interpreted as a confidence limit in the following form. The probability of having x in the interval $(\mu-\sigma, \mu+\sigma)$ is 0.68, and doubling the interval increases this probability to 0.95. In this manner, when we say 'we are within one standard deviation', this means a mathematical probability of 68 p.c.

1.3 Expectation values

Let g(x) be an arbitrary function of the random variable x , this random variable corresponding to a probability distribution function f(x) defined in the interval (a,b). Arbitrary means that g(x) must not satitify any positivity condition, i.e., it may have any shape, provided the integrals to be defined below are well defined.

The expectation value of g is defined as

$$\langle g \rangle = \int_a^b g(x)\ f(x)\ dx \qquad (1.4)$$

Some particular forms of g(x) have an special interest:

i) MEAN value of the distribution, when $g(x)=x$,

$$\mu = \int_a^b x\ f(x)\ dx \qquad (1.5)$$

ii) CENTRAL MOMENTS,. i.e., moments of the distribution with respect to the mean value,

$$g(x) = (x-\mu)^n$$

$$\mu_n = \int_a^b (x-\mu)^n\ f(x)\ dx \qquad (1.6)$$

The second moment μ_2 is called the variance of the probability distribution, and its square root is called the standard deviation which usually is represented by the greek letter σ .

$$\sigma = \sqrt{\mu_2} \qquad (1.6a)$$

We may now recall the example of eq. (3) referring to the normal distribution. The parameters μ and σ introduced there correspond to the mean and standard deviation of this distribution, respectively.

iii) CHARACTERISTIC FUNCTION, when $g(x)=\exp(itx)$,

$$\phi(t)= \int_a^b \exp(itx)\ f(x)\ dx \qquad (1.7)$$

The n-th derivative of the characteristic function is proportional to the n-th (non-central) moment

$$\left\{\frac{d^n}{dt^n}\ \phi(t)\right\}_{t=0} = i^n \langle x^n \rangle \qquad (1.8)$$

There is an important theorem regarding the moments of a given distribution: under quite mild conditions on the distribution function, the set of moments uniquely determines the probability distribution.

1.4 The Central Limit theorem

Let x be a random variable with probability distribution function f(x). Consider a new random variable z

$$z = (x_1+x_2+ \ldots +x_n) / n \qquad (1.9)$$

i.e., the average of n random values of the variable x. The question is to determine the probability distribution function of this new variable. It can be computed using the following formula

$$g(z)=\int dx_1\ dx_2 \ldots\ dx_n\ f(x_1)f(x_2)\ \ldots f(x_n)\ \delta\left(\frac{x_1+x_2+..+x_n}{n} - z\right) \qquad (1.10)$$

where δ represents Dirac's delta function. Using the integral representation of the delta function, shifting the integration variables with the mean value and reordering the integrals, eq. (10) is converted into

$$g(z)= \frac{1}{2\pi}\int_{-\infty}^{\infty} dt\ e^{-i(z-\mu)t} \left\{\int f(x)\ e^{it(x-\mu)/n}\ \ dx\right\}^n \qquad (1.11)$$

This gives an explicit way of evaluating the new distribution function, which consists in determining the characteristic function of the original distribution with a scaled and shifted variable, then rising it to the power n and finally computing the inverse Fourier transform. In general eq. (11) is not simple to evaluate. However, in the case of large n we may approximate this equation by expanding the second exponential of (11) and keeping only the dominant terms,

$$\int f(x)\ \exp(it(x-\mu)/n)\ dx \approx \int f(x)dx \left\{1 + it(x-\mu)/n - t^2(x-\mu)^2/2n^2\ \ldots\right\}$$

The second term gives a zero contribution, and the third term is related to the variance of the distribution, i.e.,

$$\int f(x)\ \exp(it(x-\ )/n)\ dx \approx\ 1 - t^2\sigma^2/2n^2 \ldots$$

Substituting in eq. (11) and carrying out the remaining integral there results the following value for the new distribution function,

$$g(z) = \frac{1}{\sqrt{2\pi}} \frac{1}{(\sigma/\sqrt{n})} \exp \quad - \frac{(z-\mu)^2}{2(\sigma/\sqrt{n})^2} \tag{1.12}$$

i.e. we obtain the normal distribution, cf. eq. (3), with a mean which is the mean of the original distribution, and a variance equal to the variance of the original distribution divided by n.

The above statement is known as <u>central limit theorem</u>. According with this theorem, the average of <u>n</u> samples is an <u>unbiased</u> estimator of the mean of the distribution. Moreover, its variance decreases with 1/n, and its standard deviation, i.e., the error in the determination of the mean, decreases with the square root of n. This is the idea which governs Montecarlo method. Note that in order to halve the error we must multiply the number of samples times 4.

The central limit theorem can be generalized to other types of expectation values. In general, if we have the expectation value

$$m = \langle h \rangle = \int f(x)\, h(x)\, dx \tag{1.13}$$

and define

$$s^2 = \int f(x) \,(\, h(x)-m)^2\, dx, \tag{1.14}$$

and consider the random variable

$$z = (\, h(x_1)+h(x_2)+...+h(x_n)\,)/n \tag{1.15}$$

then it turns out that z is distributed with a normal distribution of mean m and variance s^2/n.

1.5 Summary

This section describes some simple statistical concepts basic for the Montecarlo method. The most relevant concept of this section is Central Limit theorem, which provides an algorithm to determine empirically average values of arbitrary functions with respect to a given probability distribution function f(x). In other words, it permits the evaluation of definite integrals. The estimator is the average value of the function, eq. (15). To carry out in practice this algorithm it is necessary to devise a method of sampling a random variable x accordingly with a prescribed probability distribution function f(x). This will be the subject of next chapter.

Along this chapter we have made reference only to univariate distributions. All these ideas may be extended to multivariate probability distributions $f(x_1,x_2,...x_n)$. When a multivariate distribution can be written as a product of univariate distributions, i.e.,

$$f(x_1,\, x_2,...,x_n) = f_1(x_1)\; f_2(x_2)\; ...\; f_n(x_n)$$

that variables are called <u>uncorrelated</u>.

This section does not include references. All concepts discussed here may be found in any elementary textbook of statistics.

2. ON THE GENERATION OF RANDOM NUMBERS WITH PRESCRIBED DISTRIBUTION FUNCTIONS

A task of primary interest in Montecarlo work is the generation of random numbers which follow a given probability distribution function. In this chapter we are going to consider the basic methods, which unfortunately will not solve the general problem. The methods considered here are the change of variables, the sum or product of random numbers and the acceptance rejection method. The powerful Metropolis algorithm will be studied latter on, as well as the composition method.

To complement this section the reader may consult Chapter 3 of Hammersley and Handscombe book /1/, or the article of Zelen and Severo /2/.

2.1 The uniform distribution U(0,1)

Usually it is a routine of your computer, callable from many programming languages (Fortran, Pascal, Basic,...) which is based in a congruential relation

$$N_i = (a\ N_{i-1} + b)\ (\text{modulus } M)$$
$$x_i = N_i/M \qquad (2.1)$$

In the above equations N_i is called a seed, and is an integer number, and x_i is the random number in the interval (0,1). Note that 1 can never be reached, but 0 is included in the generation interval. The constants a, b and M are chosen so as to satisfy the following three requirements:

i. The resulting distribution is uniform, and the random numbers generated are uncorrelated.
ii. The period of eq. (1) is as long as possible, and
iii. The algorithm is fast.

A quite complete analysis of congruential methods may be found in Ref. /3/. Currently, the number M is determined by the number of bits of the integer word of the computer, i.e., is the larger unsigned integer. In this form the modulus operation is automatically carried out by discarding the overflow digits in the previous integer operation. Analogously, the division N/M is never carried out, being just a shift of bits and the corresponding adjustment of the floating exponent.

Randomness cannot be easily studied. One should analyze the posssible correlations between N_i and N_{i+s}, s arbitrary, as well as the case in which random numbers are considered by pairs, triplets and so on. The routines are empirically tested. Binder /4/ describes several problems of the common computer routines, and also some simple ways to improve the quality of the sequence.

The generation of uniformly distributed random numbers is very interesting because it is the basic algorithm to generate random numbers corresponding to other probability distributions. With this routine all Montecarlo work can be carried out.

From $x \in U(0,1)$ one can obtain $z \in U(a,b)$ by the transformation

$$z = a + (b-a)\ x \qquad (2.2)$$

2.2 Change of variables

Let x correspond to a distribution f(x). Define a new variable $y=h(x)$. The question is to determine the distribution function g(y) of the new variable y.

We must have

$$f(x)\ dx = g(y)\ dy$$

so that

$$g(y) = f(\ h^{-1}(y))\ \ dh^{-1}(y)/dy \qquad (2.3)$$

where h^{-1} is the inverse function of h. When f(x) is the uniform distribution we should only consider the derivative term,

$$g(y) = dh^{-1}(y)/dy \qquad (2.4)$$

Example 1: Let $y=x^2$. The inverse function is the square root, and the distribution of y is

$$g(y) = d\ y^{1/2}/dy = 1/2y^{1/2} \qquad (2.5)$$

The new variable is also defined in the interval (0,1).

We are currently interested in the inverse problem, i.e. g(y) of eq. (4) is known and we seek for the change of variables $h^{-1}(y)$. Integrating eq. (4) we get

$$h^{-1}(y) = \int^{y} g(x)\ dx\ \ = G(y) \qquad (2.6)$$

so that the function which defines the change of variables is the inverse of the cumulative distribution function.

For this procedure to be useful we then require i) to be able to compute the cumulative distribution function, and ii) to be able of inverting it. These are two strong requirements which can be fulfilled only in few cases.

Example 2: Consider the example 1. The cumulative distribution is

$$G(y) = \int_0^{y} dx\ /2\ x\ \ = y^{1/2}$$

and the change of variables is $y=x^2$.
Sampling from this distribution is simple: draw $x \in U(0,1)$ and

square it.

2.3 The normal distribution

In the case of the normal distribution its cumulative distribution function is the complementary error function, which cannot be exactly inverted. This problem may be circunvected by considering the bivariate normal distribution

$$f(x_1,x_2) = \frac{1}{2\pi} \exp \left\{- \frac{1}{2} (x_1^2 + x_2^2)\right\} \qquad (2.7)$$

and change to polar coordinates

$$x_1 = r \cos\varphi$$
$$x_2 = r \sin\varphi \qquad (2.8)$$

so that the distribution function in terms of the new variables is

$$g(r,\varphi) = r \exp(-r^2/2) /2\pi \qquad (2.9)$$

i.e., $r \exp(-r^2/2)$ corresponding to r and uniform $U(0,2\pi)$ corresponding to φ. Now the cumulative functions can be computed and inverted, and the sampling algorithm is

$$x_1 = (-2 \log z_1)^{1/2} \cos(2\pi z_2)$$
$$x_2 = (-2 \log z_1)^{1/2} \sin(2\pi z_2) \qquad (2.10)$$

where z_1 and z_2 are $U(0,1)$ random variables. In this form one obtains pairs of normally distributed random numbers. Currently both numbers can be taken, but this may amplify possible correlations already present in the uniform distribution algorithm. This may be avoided by considering x_1 or x_2 with probability 1/2 each. The final sampling algorithm is as follows:

i. Draw z_1 and z_2 form $U(0,1)$
ii. Compute x's with eqs. (10)
iii. Draw z_3 from $U(0,1)$. If it is larger than 1/2 thenconsider x_1, otherwise consider x_2.

2.4 Sum and product of random variables

Consider x and y corresponding to probability distribution functions $f(x)$ and $g(y)$, respectively. The distribution corresponding to the new variable

$$z = x+y \qquad (2.11)$$

is given by

$$h(z)=\int f(x)\, dx \quad g(y)\, dy \quad \delta(x+y-z) \qquad (2.12)$$

i.e., the convolution

$$h(z=x+y) = \int f(x)\, g(z-x)\, dx \qquad (2.13)$$

In a similar way we may obtain the distribution of the product of two random variables, $z=xy$, just by changing the argument of the delta function in eq. (12) to $(xy-z)$. The result is

$$h(z=xy) = \int f(x)\ g(z/x)\ dx\ /\ x \tag{2.14}$$

Example 3: Let both $f(x)$ and $g(x)$ be the uniform distribution $U(0,1)$. Then xy is defined in the range $(0,1)$ also.We may evaluate eq. (14) by means of the step function, i.e.

$$h(z) = \int_0^1 dx\ \theta(z/x)\ \theta(1-z/x)\ /\ x.$$

The product of the step functions is only different form zero only when $x>z$, so that the integral is efectively extended from z to 1. The distribution of the product is then $-\log(z)$.
Similar steps may be followed in the case of the sum $z=x+y$.Now the range is $(0,2)$ and the distribution is

$$\begin{array}{lll} z & \text{when} & z<1 \\ 2-z & \text{when} & z>1. \end{array}$$

Example 4: See Ref. /5/
Consider now the distribution $h(z,n)$, where

$$z = -\log(x_1 x_2 \dots x_n)\ ,\ x_i \in U(0,1) \tag{2.16}$$

The new variable is defined in the range $(0,\infty)$. The form of the distribution may be obtained by induction. In the case $n=1$, i.e., $z=-\log(x)$ we can use the change of variables method, sect. 2.2, to obtain

$$h(z,1) = \exp(-z) \tag{2.17}$$

When $n=2$, z corresponds to the sum of two random variables which are distributed exponentially, eq. (17). Using eq. (13) there results

$$h(z,2) = z\ \exp(-z) \tag{2.18}$$

For arbitrary n we guess

$$h(z,n) = z^{n-1}\ \exp(-z)\ /\ (n-1)! \tag{2.19}$$

and this can be confirmed by computing the convolution of $h(z,n-1)$ with $h(z,1)$. Equation (19) is the Gamma distribution, and eq. (16) gives the sampling algorithm. This example is quite interesting, because this distribution appears in the determination of the free Green function of a many body system.

2.5 Acceptance-Rejection method

This method was invented by von Neumann /6/. Let $f(x)$ be defined in a finite interval, and bounded by the value M. Random numbers corresponding to this distribution can be obtained in the following form:

i. Draw x from U(a,b)
ii. Draw p from U(0,M)
iii. If $p \leqslant f(x)$ then accept x as the new random number.
Otherwise, reject both x and p and repeat the algorithm from the beginning.

These three steps may be condensed in a single sentence: draw x from U(a,b) and accept it with probability f(x)/M.

This method is illustrated in figure 1, which shows a case of acceptance and a case of rejection. The proof of the algorithm is quite simple, just write the previous statement mathematically,

$$\int_a^b \frac{1}{(b-a)}\, dx \int_0^M \frac{1}{M}\, dp\ \delta(x-z)\ \theta(f(x)-p) = \frac{1}{M(b-a)}\, f(z) \qquad (2.20)$$

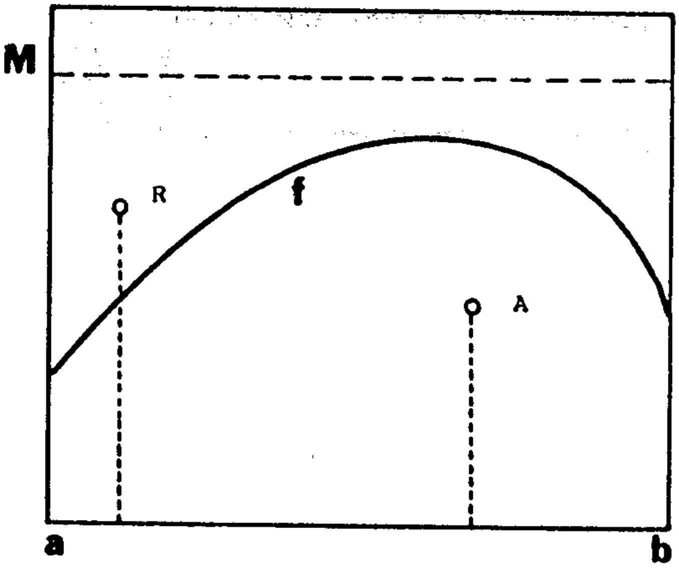


Figure 2.1: The acceptance-Rejection method. The figure shows one point accepted (A), and one rejected (R).

The quantity 1/(b-a) is the uniform distribution of x, U(a,b). The delta function says the result of sampling z is x. The step function means that only when f(x) is greater than p the value of x will be accepted. The integral is trivially carried out and the result is in the rhs of eq. (20). We obtain the desired distribution, but divided by M(b-a), which is always greater than 1. The lack of normalization of the resulting distribution reflects the loss of points due to rejections. For N trials, only N/M(b-a) will be accepted. To have high efficiency M should be as small as possible, namely the maximum value of f(x) in the interval (a,b).

The acceptance-rejection method may be improved so as to have maximum efficiency, i.e., so as to obtain on the average N points from N trials. The improved algorithm is the following:

i. Draw x from U(a,b)

ii. Compute f(x)

iii. Draw an integer n with mean value f(x)/M, where M=1/(a-b). The way of doing it is

$$n = \mathrm{Int}(z + f(x)/M)$$

where z is from U(0,1).

iv. Count n times the value x, i.e., if n=0 then reject it, if n=1 count it once, and so on.

The trials rejected are compensated by the trials counted more than once, and on the average we will end up with N random numbers after N trials. Note that now M is not an upper bound to the distribution, but 1/(b-a). This improved algorithm is illustrated in figure 2.

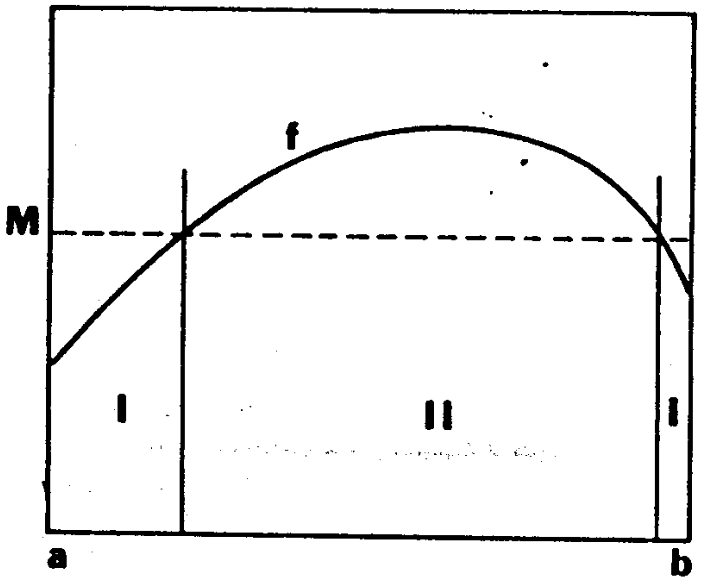


Figure 2.2: Improved acceptance-rejection method.

Points in regions labelled I are not counted, or are counted only once. Points in region labelled II are counted once or twice

2.6 Empirical determination of probability distributions

The various algorithms above described may be easily used and tested by means of even a home computer. The method is quite simple. Just generate a long set of random numbers using the appropriate sampling formula and accumulate the results in the form of an histogram, i.e. count an plot the number of times the generated random numbers are within x_i and x_{i+1}. These values of x define the intervals used to define the histogram. To compare with the theoretical form of the distribution the obtained numbers must be scaled. The number of times a random number is in the interval (x_i, x_{i+1}) is given by

$$N_s \int_{x_i}^{x_{i+1}} f(x)\, dx \approx N_s (x_{i+1} - x_i)\, f(x_i)$$

where N_s is the total number of samples.

Table 2.1. Examples of probability distributions, the sampling algorithms and results. The first column refers to figure (2.1)

Figure	Distribution	Range	Mean	Variance	Sampling	Empirical	
						Mean	Variance
A	1/2 x	(0,1)	.333	.0888	$x = z^2$	.344	.0911
B	exp(-x)	(0,∞)	1	1	x=-log(z)	1.01	1.057
C	2x	(0,1)	.666	.0555	$x=z^{1/2}$	.671	.0526
D	Normal	(-∞,∞)	0	1	See text	-.03	.9767

```
C
C       EMPIRICAL DETERMINATION OF P.D.F.
C
        DIMENSION IBIN(20)
        XMIN=0.0                            ! LOWER LIMIT FOR HISTOGRAM
        XMAX=1.0                            ! UPPER LIMIT
        STEP=0.05                           ! SIZE OF BIN
        NBINS=(XMAX-XMIN)/STEP              ! NUMBER OF BINS
        NSAMPLES=1000
        AVERAGE=0.0
        SQAVERAGE=0.0
        DO 10 I=1,NSAMPLES
        X=RAN(K)**2                         ! SAMPLING FORMULA
                                            ! CHECK CALLING OF RANDOM ROUTINE
        AVERAGE=AVERAGE+X                   ! COMPUTE MEAN
        SQAVERAGE=SQAVERAGE+X**2            ! COMPUTE VARIANCE
        IF (X .GT. XMAX) GOTO 10            ! DISCARD IF OUTSIDE RANGE
        IX=(X-XMIN)/STEP+1
        IBIN(IX)=IBIN(IX)+1                 ! ACTUALIZE HISTOGRAM
10      CONTINUE                            ! END OF SAMPLING
        XMEAN=AVERAGE/NSAMPLES
        VARIANCE=SQAVERAGE/NSAMPLES-XMEAN**2
C
C       PLOT AND PRINT RESULTS
C
        DO 20 I=1,NBINS
20      PRINT*,I,IBIN(I)
        PRINT*,'    MEAN=',XMEAN
        PRINT*,'VARIANCE=',VARIANCE
        END
```

Figure 2.3 . A FORTRAN program to determine empirically the shape of a probability distribution function.

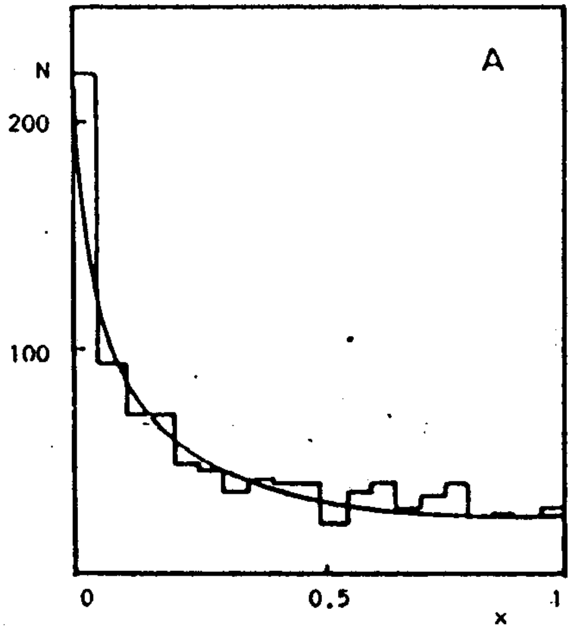


Figure 2.4.a: The distribution $1/\sqrt{x}$

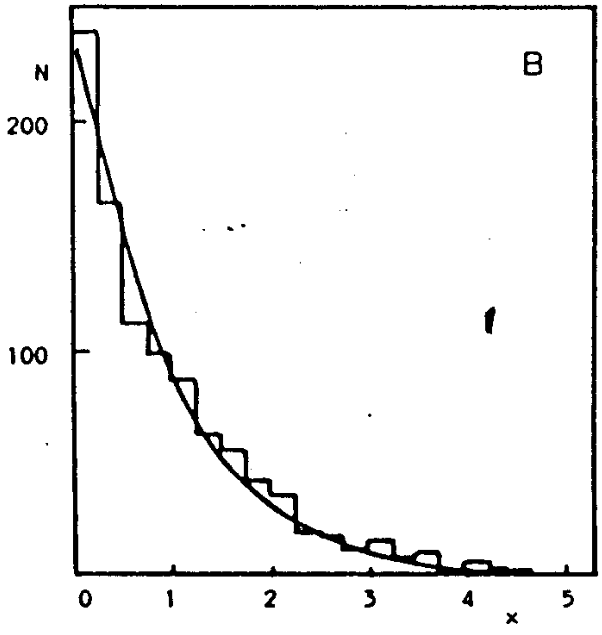


Figure 2.4.b: The exponential distribution exp(-x)

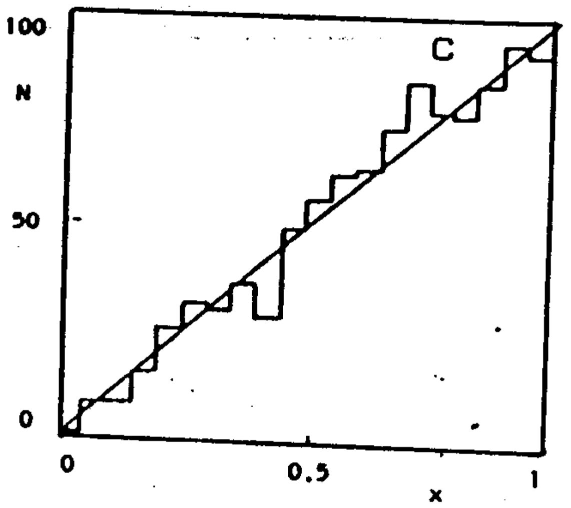


Figure 2.4.c: The triangular distribution 2x

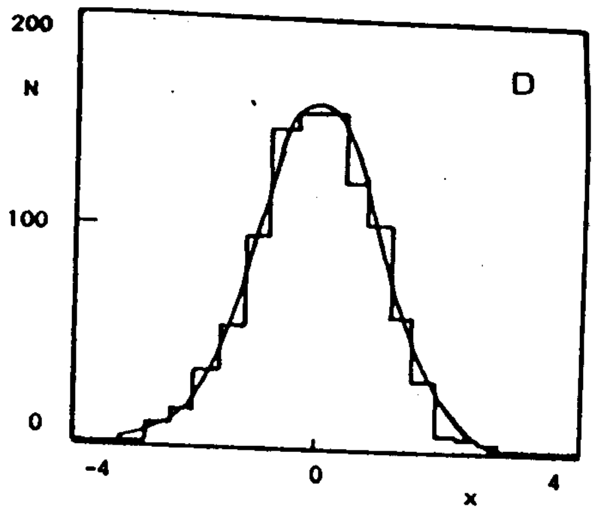


Figure 2.4.d: The normal distribution

This process is exemplified in the FORTRAN program presented in Figure 2.3, which do not requires further explanations because of its simplicity. Using this program the distributions listed in Table 2.1 have been sampled, using a total of 1000 points. The global parameters corresponding to these distributions are compared in Table 1 with their exact value, and Figures 4.a to 4.d show the corresponding histograms.

2.7 Summary of methods

1. The uniform distribution is an algorithm to be found in the system library of the computer or in a general purpose Fortran library.

2. The inverse method can be used when the cumulative distribution function can be both computed and inverted.

3. The acceptance-rejection method is useful in bounded, finite range distributions.

4. A clever use of sums, products and quotients of uniform random numbers may produce some simple sampling algorithms in special cases.

5. There remain still a large amount of sampling methods which cannot be classified. For example, the algorithm $x=\max(x_1,x_2)$, both x_1 and x_2 uniformly distributed random numbers, generates the triangular distribution 2x. If x_1, x_2, and x_3 are U(0,1), and they are ordered, then the distribution of $x=x_2$, the middle value, is 6x(1-x).

6. This section has not covered two very important methods which are the basis of the Variational and Green Functions Montecarlo theories. They are the Metropolis algorithm and the composition method, which will fully described below.

3. MONTECARLO QUADRATURE AND METROPOLIS ALGORITHM

3.1 Crude Montecarlo quadrature

We are interested in the computation of the integral

$$I = \int_a^b h(x)\, dx \quad . \tag{3.1}$$

where h(x) is an arbitrary ~~(positive or negative)~~ function, provided the integral is defined.

Let $(x_1, x_2, \ldots, x_N)$ be a set of random numbers uniformly distributed in the interval (a,b). Because of central limit theorem we can write

$$I = \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} h(x_i) \tag{3.2}$$

The variance may also be determined

$$\sigma^2 = \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} h^2(x_i) \quad - I^2 \tag{3.3}$$

If the sampling is limited to N random numbers one usually writes the result of the integral as

$$I \pm (\sigma^2 / N)^{1/2} \tag{3.4}$$

where the error corresponds to the 68% of confidence.

The error of the quadrature depends on two quantities, the variance of the function h(x) to be integrated and the number of samples. Note that the variance depends <u>only</u> on the function h(x),

$$\sigma^2 = \int_a^b (h(x) - I)^2 dx \tag{3.5}$$

and has no relation with the number of samples or the sampling method. As a consequence, to diminish the computational error in this crude method the only way is to increase the number of samples. Specifically, to halve the error we must work four times more. The variance measures the quadratic deviation of our function h(x) from its mean value I. Rapidly varying functions will have a large variance and poor precision.

To improve the efficiency (precision versus work) of crude Montecarlo several techniques have been developed, like importance sampling, stratified sampling, control variates, anatitetic variates and so on. Ref. /1/ gives a quite complete description and analysis of all these techniques. We will concentrate here on importance sampling because of its direct relation with the techniques in the many body problem.

3.2 Importance sampling

The idea of importance sampling is to modify the function to be sampled, as well as the sampling algorithm, so as to significantly decrease the variance of the resulting function. Consider a probability distribution function f(x) defined in the same interval (a,b) of the quadrature. Equation (1) may be rewriten as

$$I = \int_a^b (h(x)/f(x)) \; f(x)\, dx \qquad (3.6)$$

and the Montecarlo estimate of the integral can be obtained by drawing random numbers from the probability distribution f(x) and computing the average

$$I = \lim \frac{1}{N} \sum (h(x_i)/f(x_i)) \qquad (3.7)$$

The new value of the variance is

$$\sigma^2 = \int_a^b (h(x)/f(x))^2 \; f(x)\, dx \;- I^2 \qquad (3.8)$$

With a judicious choice of f(x) me may achieve a small variance, so that the extra work required by the new formula (3.7) may be compensated by the need of a smaller number of samples.

Example 3.1: The integral of the function h(x)= (exp(x)-1)/(e-1) , in the (0,1) interval, is 0.418, and its variance is 0.082. If the importance sampling function f(x)=2x is used, the new variance is lowered to 0.0027. Of course, the value of the integral remains the same.

3.3 Markov chains

The main problem in the importance sampling technique is the determination and use of the importance sampling function. To lower significantly the variance, the importance sampling function should mimic the function h(x) to be integrated. This loose criterium does not help too much in chosing the required distribution function. On the other hand, there remains still the problem of carrying out the required sampling with this function. Both problems are formidable in multivariate integrals. In many cases, the importance sampling function is dictated by the mere structure of the physical problem at hand, as in the case of the variational many body problem. In this case, the importance sampling function is the trial wave function squared. However, there remains still the problem of sampling from a normally very complicated distribution function. To solve this problem, Metropolis, Rosenbluth, Rosenbluth, Teller and Teller /7/ have devised a stochastic algorithm, i.e., a random walk in the multivariate coordinate space which generates asymptotically the desired set of random numbers. This algorithm is known with the name of Motropolis, or the acrostic $M(RT)^2$.

This random walk is of the type known as Markov chain or Markov process. Consider a system which can be in a discrete set of states S_1, S_2... S_N. The evolution of this system is governed by a clock, so that at every jiffy the system jumps from its current state to any other state of the set, itself included. This jump is characterized by a transition probability matrix p_{ij}, which represents the probability ot jumping from the state i to any other state j. Note that p_{ii} is also included, and that these matrix elements must fulfill the following conditions:

$$0 \leqslant p_{ij} \leqslant 1 \qquad (3.9)$$

$$\sum_j p_{ij} = 1 \qquad (3.10)$$

In addition, it is required that the transition from i to j is independent on the previous occupied states during the evolution.

Equations (9) and (10) are obvious. Equation (9) says that the probability of jumping from i to j is a positive number less or equal than 1. Equation (10) is the sum rule corresponding to these probabilities. Note that the sum is over the second index, actually there are no restrictions on the corresponding sum extended to the first index. A matrix which satisfies both (9) and (10) is called a stochastic matrix, and the process related to it is called a Markov chain.

The generalization to the case of the continuum is called a Markov process. We should use now x to label the states, and define a transition density p(x,x') with the properties

$$\begin{aligned} &p(x,\ x') \geqslant 0 \\ &\int p(x,\ x')\ dx' = 1 \end{aligned} \qquad (3.11)$$

The following question emerges naturally. After a very long walk, which is the probability P_k of having passed on some state S_k? . More precisely, assume we carry out N jumps, and let N_k be the number of stops at S_k. The searched probability is $P_k = \lim N_k/N$.

For this question to have sense we must add two new conditions to the stochastic matrix

i. The random walk must be endless, i.e., there is none state with $p_{ii}=1$. If that state would exist, then our walker would be trapped at it. This kind of state is called an absorbing wall. Terminating random walks have a deep interest in many physical problems, like shielding of radiation, but we are not interested at them here.

ii. The chain is irreducible. Probably the opposite concept is easier to understand: if we can classify the states in two subsets so that there cannot be a transition from one subset to the other, then the chain is termed reducible. A closely related word to this concept is ergodicity.

3.4 Solution of the direct problem

The algebraic determination of the probabilities P_k is quite simple. The probability of arriving to a state S_i is the product of the probability of being previously in another state S_k times the transition probability, i.e.,

$$P_i = \sum_k P_k \; p_{ki} \qquad (3.12)$$

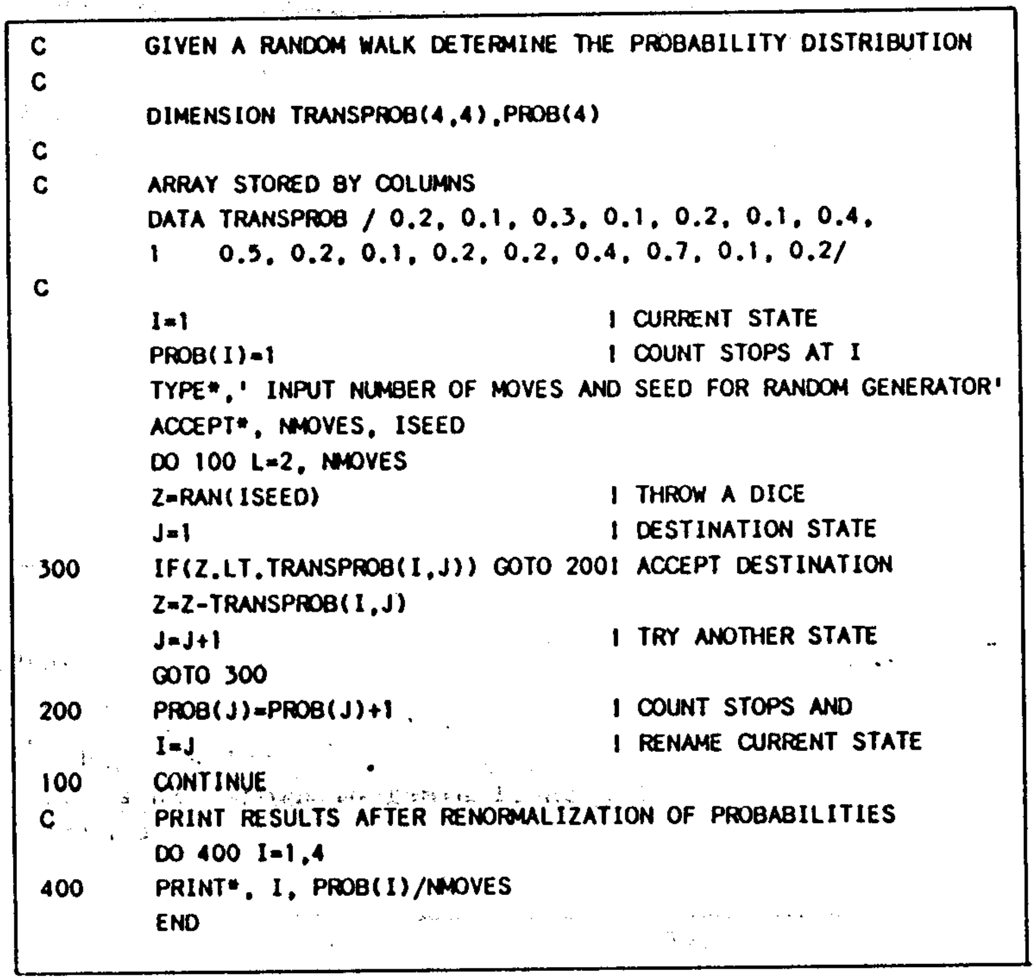
This is a set of homogeneous linear equations, which as expected are not independent, as it can be checked by summing up over the free index i. It should be supplemented by the normalization condition

$$\sum P_i = 1 \qquad (3.13)$$

and the system can now be solved.

There is however an alternative way of facing up this problem, just by walking along the random walk process. Instead of describing the method we present in Figure 1 the corresponding Fortran program in a simple case: start in a given state (I=1) and do NMOVES jumps. Sentence labelled 300 decides the destination of a given jump by comparing with a U(0,1) random number.

The results corresponding to several values of the total number of moves are shown in Table 1, as well as the exact results. Certainly the method works.



```
C       GIVEN A RANDOM WALK DETERMINE THE PROBABILITY DISTRIBUTION
C
        DIMENSION TRANSPROB(4,4),PROB(4)
C
C       ARRAY STORED BY COLUMNS
        DATA TRANSPROB / 0.2, 0.1, 0.3, 0.1, 0.2, 0.1, 0.4,
        1    0.5, 0.2, 0.1, 0.2, 0.2, 0.4, 0.7, 0.1, 0.2/
C
        I=1                             ! CURRENT STATE
        PROB(I)=1                       ! COUNT STOPS AT I
        TYPE*,' INPUT NUMBER OF MOVES AND SEED FOR RANDOM GENERATOR'
        ACCEPT*, NMOVES, ISEED
        DO 100 L=2, NMOVES
        Z=RAN(ISEED)                    ! THROW A DICE
        J=1                             ! DESTINATION STATE
300     IF(Z.LT.TRANSPROB(I,J)) GOTO 200! ACCEPT DESTINATION
        Z=Z-TRANSPROB(I,J)
        J=J+1                           ! TRY ANOTHER STATE
        GOTO 300
200     PROB(J)=PROB(J)+1               ! COUNT STOPS AND
        I=J                             ! RENAME CURRENT STATE
100     CONTINUE
C       PRINT RESULTS AFTER RENORMALIZATION OF PROBABILITIES
        DO 400 I=1,4
400     PRINT*, I, PROB(I)/NMOVES
        END
```

Figure 3.1. A random walk solution of a system of linear equations, equations (12) and (13).

Table 3.1: A random walk solution of a system of linear equations.

	EXACT	100 Moves	1000 Moves	10000 Moves
P(1)	0.14859	0.10	0.149	0.1494
P(2)	0.31325	0.36	0.310	0.3088
P(3)	0.16867	0.16	0.172	0.1710
P(4)	0.36984	0.38	0.369	0.3708

We have discovered a surprising form of solving systems of linear equations. Many other simple and surprising examples of random walks may be found in the book of Martin Gardner, Ref. /8/. This algorithm has been generalized /1/ to a class of linear equations of the form

$$x_i = a_i + \sum_j H_{ij} x_j \qquad (3.14)$$

where the euclidean norm of the matrix H is less than one, i.e.,

$$\max_i \left\{ \sum_j H_{ij} \right\} < 1 \qquad (3.15)$$

but otherwise is arbitrary. The formal solution of (14) is

$$x = \frac{1}{1-H} a \qquad (3.16)$$

and condition (15) guarantees that the series expansion of (16),

$$x = (1 + H + H^2 + H^3 + \ldots) a \qquad (3.17)$$

converges. The generalized algorithm consists in constructing this series by means of a random walk.

In the case of the continuum the equivalent to (14) is a nonhomogeneous linear integral equation. Techniques for the solution of this kind of integral equations may be found in Ref/ /9/.

3.5 The inverse problem: Metropolis algorithm

In the previous section we have solved a direct problem: given a random walk determine a probability distribution. Our problem, namely sampling random numbers with a prescribed probability distribution, is just the inverse of the previous problem. Now we know the value of P_i and we want to determine a random walk, i.e., the stochastic matrix p_{ij}. The problem is overdetermined, and the solution is not unique.

Metropolis and coworkers /7/ have determined a solution. Let q_{ij} be a stochastic and symmetric matrix, otherwise arbitrary. Then our random walk matrix is given by

$$p_{ij} = q_{ij} \qquad \text{if } P_i > P_j \text{ , } i=j \qquad (3.18)$$

$$p_{ij} = q_{ij} P_j/P_i \qquad \text{if } P \leq P \text{ , } i=j \qquad (3.19)$$

$$p_{ii} = q_{ii} + \sum_k{}' q_{ik}(1-P_k/P_i) \qquad (3.20)$$

The value of p_{ii}, eq. (20) has been fixed so as to fulfill condition (10) of the stochastic matrix. The prime on the summation symbol means that only states which satisfy P_k P_i are considered.

The meaning of above equations is the following. Assume we are sitting at S_i. The choose a trial state S_j with probability q_{ij}. If the probability of the trial state is greater that our current probability, then the move is accepted. This is eq. (18). Otherwise, the trial state is accepted with probability P_j/P_i, which is now smaller than one. In the case the new state is not accepted, then the new state (i.e., the new random number) is the old state S_i, which is then counted twice.

The proof of Metropolis form is quite simple. Actually we should only check that this p_{ij} satisfies the set of equations (12), because the stochastic conditions (9) and (10) are satisfied by construction. An intermediate step is to show that this transition matrix satisfies the microreversibility or detailed balance condition

$$P_i \, p_{ij} = P_j \, p_{ji} \qquad (3.21)$$

There is no summation on repeated indexes in eq. (21). Once this condition is checked, eq. (12) turns out just by summing up in (21) on index i. We leave the proof of (21) as an exercise to the reader.

Metropolis algorithm can be generalized to the case of a continuous probability distribution p(x). The moves corresponding to the auxiliary matrix q_{ij} can now be carried out by means of a uniformly distributed displacement within a segment line of length D centered at the current position x.

The algorithm is as follows. Assume we are at point x, then

i. Try a move to $x' = x + D\,(2z-1)$, where z is from U(0,1).

ii. If $p(x') > p(x)$ then go to step iv.

iii. If we arrive here is because p(x') is smaller than p(x). Then sample a new random number z from U(0,1) and do the following test:

```
    IF $p(x')/p(x) > z$ then go to iv,
    ELSE
    put $x' = x$, i.e., remain at the old position.
```

iv. Put x' in the list of random numbers, rename x' to x and go back to step i.

Distance D is arbitrary, the only condition on the auxiliary matrix q being its symmetry. When D is very small, the numbers of acceptances in steps ii and iii is large, but a lot of moves is necessary to cover the space and the successive random numbers are strongly correlated. On the contrary, when D is large the number of rejections is large, so that we may spend a long time at every position. It is usually admitted that the appropriate value of D should give a ratio of acceptances between 50% and 70%.

The same algorithm can be applied to the case of multivariate distributions. The most common problem in physics involves a probability distribution which depends on the coordinates of N particles, i.e., 3N variables. The random walk is ussually carried out in N steps, moving in turn each particle (not necesarily in the same order), and accepting or rejecting the individual motions with the usual test. After all particles have been tried, the resulting point is the new multidimensional random number. Note that some of the new coordinates may have the same value than the old coordinates.

A usual procedure to avoid correlations among the succesive positions is the so called thermalization. Metropolis algorithm is repeated a number of times before accepting the new point for sampling. This will increase the computing time. In any case, we should always have in mind that Metropolis algorithm is asymptotic.

4. VARIATIONAL MONTECARLO

4.1 General description

The usual form of a many body hamiltonian is

$$H = - \frac{h^2}{2m} \sum_i \nabla_i^2 + \sum_{i<j} V(i,j) \qquad (4.1)$$

involving one body kinetic energy terms and a two body potential energy interaction. The two body interaction may depend on the relative distance between the particles as well as on spin or isospin degrees of freedom.

One of the features which makes the many body problem so complex is the fact that in almost all interesting many body systems the two body potential shows a strong repulsion at short distances, in such a form that plain perturbative methods cannot be straightforwardly applied.

A simple alternative to the exact solution of eq. (1) is to compute a variational upper bound to the ground state energy using a trial function which takes into account appropriately the repulsive core of the two body interaction. The simplest form is known with the name of Jastrow /10/, and consists in two pieces, a model wave function made up of single particle orbitals and a correlation factor which is the product of all pairs of two body correlations.

Some of the physically interesting forms are:

i. A finite system of bosons (drops)

$$\Psi_T = \prod_i (r_i - R) \prod_{i<j} f(r_{ij}) \qquad (4.2)$$

The first term clusters all particles around their center of mass

R, and the second is the correlation factor.

ii. An infinite system of bosons (a liquid or a gas)

$$\Psi_T = \prod_{i<j} f(r_{ij}) \qquad (4.3)$$

In this kind of problem the product is extended to a finite number of particles which stay in a simulation box, and afterwards periodically extended to cover all space.

iii. A bosonic crystal

$$\Psi_T = \prod_{i} (r_i - A_i)\cdot \prod_{i<j} f(r_{ij}) \qquad (4.4)$$

where A_i are the lattice sites of the crystalline structure, and the single particle wave functions tie each particle to a lattice site. Again this wave function is limited to a finite number of particles and periodically extended to all space.

Systems of fermions may be described in a similar way, the main change being the presence of a Slater determinant to account for the antisymmetric character of the wave function.

More general forms of trial functions may include three body correlations, momentum dependence of the correlation factor and spin or isospin dependence. I will mainly consider the simplest forms only.

The upper bound to the ground state enegy is given by the multidimensional integral

$$E_V = \frac{\int \Psi_T^*(R)\ H\ \Psi_T(R)\ dR}{\int \Psi_T^*(R)\ \ \Psi_T(R)} \qquad (4.5)$$

where R represents the set of the 3N coordinates. When particles are spinless and the hamiltonian is a local operator, eq. (5) may be writen in a form appropriate for Montecarlo calculations. Define a <u>local energy</u> $E_L(R)$ as

$$E_L(R) = \frac{1}{\Psi_T(R)}\ H\ \Psi_T(R) \qquad (4.6)$$

and introduce a multivariate probability distribution

$$p(R) = |\Psi_T(R)|^2 / \int |\Psi_T(R)|^2\ dR \qquad (4.7)$$

Note that p(R) is always positive and normalized to 1. Then the variational energy is given by

$$E_V = \int p(R)\ E_L(R)\ dR \qquad (4.8)$$

This integral has been writen in the standard form (3.6) of Montecarlo quadrature with p(r) the importance sampling function, and $E_L(R)$ the function to be sampled. The Montecarlo evaluation of the variational energy is a simple task: draw a set of random variables $R_1, R_2, \dots R_N$ using Metropolis algorithm and obtain an estimate of the energy with the quantity

$$E_V = \frac{1}{N} \sum_i E_L(R_i) \tag{4.9}$$

with an statistical error

$$E_V = \left\{ \frac{1}{N} \sum_i E_L^2(R_i) - E_V^2 \right\}^{1/2} \Big/ \; ^{1/2} \tag{4.10}$$

Note that sampling E_L^2 is not the same that computing the expectation value of H^2, unless H is a local operator.

The form of the local energy is not unique. The contribution of the potential energy term of the hamiltonian is, of course, the potential energy. On the other hand, the kinetic energy involves the computation of derivatives and the straightforward way of carrying out these derivatives gives rise to three body terms. A clever use of integration by parts technique can produce other forms, which may be simpler than the direct form of eq. (6). In any case all these forms give the same value for the variational energy, but the value of the variance and statistical error may be different. The direct way, i.e., acting with Laplace operator on the rhs wave function, has zero variance in the limit case in which the exact wave function is used as a trial wave function. This may be a good recipe to organize the calculations, but as we have mentioned above, this originates general three body terms so that computations may be expensive.

4.2 A simple example: a nucleon bound to a gaussian potential

Consider a nucleon in the gaussian potential

$$V(r) = V_0 \exp(-(r/b)^2) \tag{4.11}$$

with V_0=-45 MeV, b=2.09 fm and a mass given by $\hbar^2/m$=41.50 MeV.fm^2.

The energy of the ground state may be obtained by numerical integration of the radial Schrodinger equation, with the result of -10.813 MeV.

A simple and quite accurate trial wave function is the 1s harmonic oscillator state

$$\Psi_T = \frac{a^{3/2}}{\pi^{3/4}} \exp\left\{-\frac{1}{2} a^2 r^2\right\} \tag{4.12}$$

the local energy being

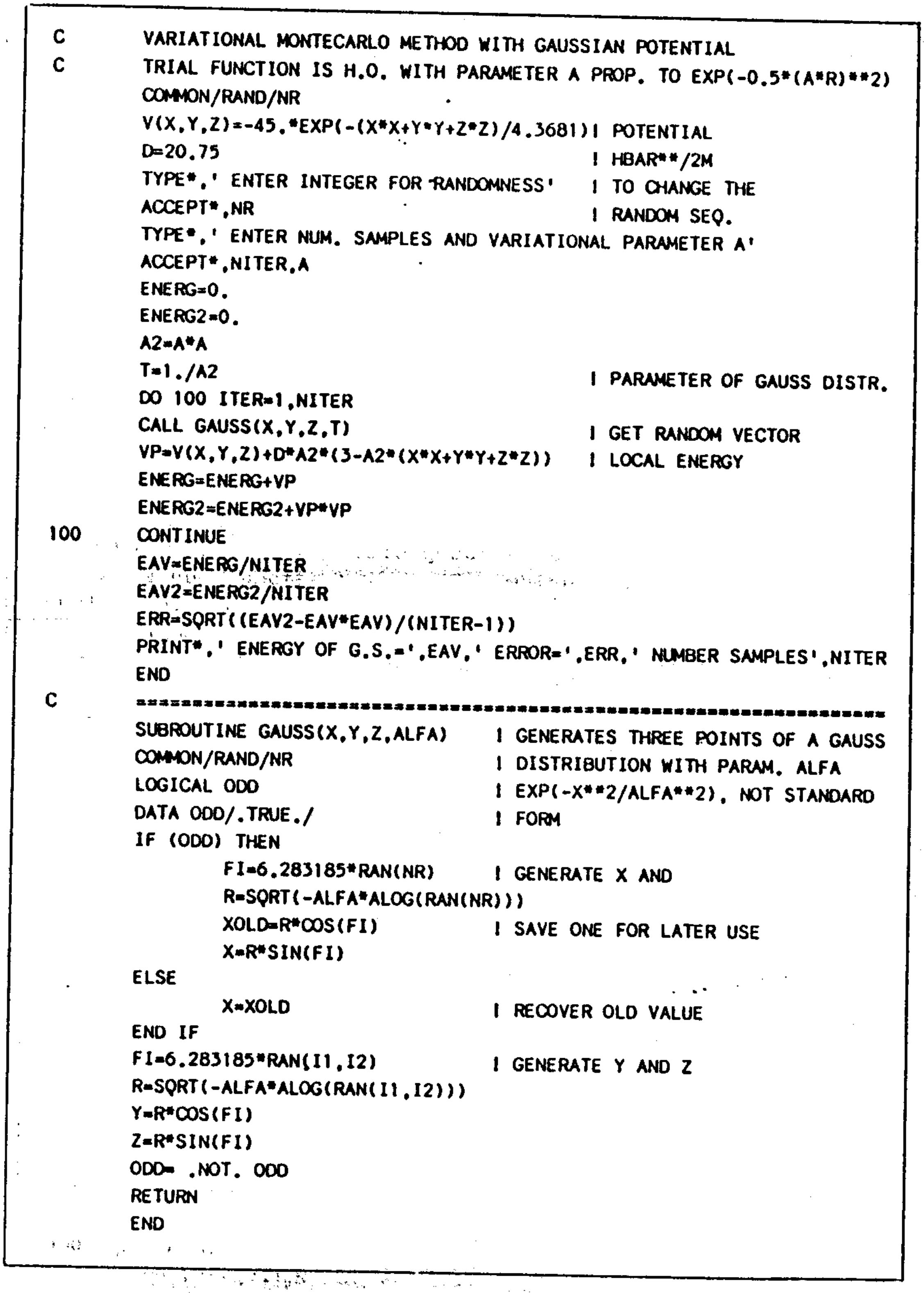
$$E_L(r) = \frac{\hbar^2}{2m} (3a^2 - a^4 r^2) + V(r) \tag{4.13}$$

The variational energy may be easily evaluated, with the result

$$E\ (a) = \frac{3\hbar^2}{4m} a^2 + V_0 \left\{ \frac{(ab)^2}{1+(ab)^2} \right\}^{3/2} \tag{4.14}$$

One can compute the expectation value of the energy in two ways.

First, by using Montecarlo quadrature algorithm with direct sampling of the wave function squared, which is a normal distribution. Secondly, we can use Metropolis algorithm to carry out the generation of random numbers.



```
C       VARIATIONAL MONTECARLO METHOD WITH GAUSSIAN POTENTIAL
C       TRIAL FUNCTION IS H.O. WITH PARAMETER A PROP. TO EXP(-0.5*(A*R)**2)
        COMMON/RAND/NR
        V(X,Y,Z)=-45.*EXP(-(X*X+Y*Y+Z*Z)/4.3681)! POTENTIAL
        D=20.75                                 ! HBAR**/2M
        TYPE*,' ENTER INTEGER FOR RANDOMNESS'   ! TO CHANGE THE
        ACCEPT*,NR                              ! RANDOM SEQ.
        TYPE*,' ENTER NUM. SAMPLES AND VARIATIONAL PARAMETER A'
        ACCEPT*,NITER,A
        ENERG=0.
        ENERG2=0.
        A2=A*A
        T=1./A2                                 ! PARAMETER OF GAUSS DISTR.
        DO 100 ITER=1,NITER
        CALL GAUSS(X,Y,Z,T)                     ! GET RANDOM VECTOR
        VP=V(X,Y,Z)+D*A2*(3-A2*(X*X+Y*Y+Z*Z))   ! LOCAL ENERGY
        ENERG=ENERG+VP
        ENERG2=ENERG2+VP*VP
100     CONTINUE
        EAV=ENERG/NITER
        EAV2=ENERG2/NITER
        ERR=SQRT((EAV2-EAV*EAV)/(NITER-1))
        PRINT*,' ENERGY OF G.S.=',EAV,' ERROR=',ERR,' NUMBER SAMPLES',NITER
        END
C       ===================================================================
        SUBROUTINE GAUSS(X,Y,Z,ALFA)    ! GENERATES THREE POINTS OF A GAUSS
        COMMON/RAND/NR                  ! DISTRIBUTION WITH PARAM. ALFA
        LOGICAL ODD                     ! EXP(-X**2/ALFA**2), NOT STANDARD
        DATA ODD/.TRUE./                ! FORM
        IF (ODD) THEN
                FI=6.283185*RAN(NR)     ! GENERATE X AND
                R=SQRT(-ALFA*ALOG(RAN(NR)))
                XOLD=R*COS(FI)          ! SAVE ONE FOR LATER USE
                X=R*SIN(FI)
        ELSE
                X=XOLD                  ! RECOVER OLD VALUE
        END IF
        FI=6.283185*RAN(I1,I2)          ! GENERATE Y AND Z
        R=SQRT(-ALFA*ALOG(RAN(I1,I2)))
        Y=R*COS(FI)
        Z=R*SIN(FI)
        ODD= .NOT. ODD
        RETURN
        END
```

Figure 4.1. Variational Montecarlo with direct sampling of the wave function squared, corresponding to the example of section 4.2.

```
C       VARIATIONAL MONTECARLO METHOD WITH GAUSSIAN POTENTIAL
C       TRIAL FUNCTION IS H.O. WITH PARAMETER A PROP. TO EXP(-0.5*(A*R)**2)
C       METROPOLIS SAMPLING
        V(R)=-45.*EXP(-R/4.3681)        ! POTENTIAL, R IS X**2+Y**2+Z**2
        FI(R,A2)=EXP(-A2*R)             ! WAVE FUNCTION SQUARED
        D=20.75                         ! HBAR**2/2M
        TYPE*,' ENTER INTEGER FOR RANDOMNESS'
        ACCEPT*,NR
        TYPE*,' ENTER NUM. MOVES and SCALE OF MOVES'
        ACCEPT*,NITER,SCAL
        TYPE*,' ENTER VARIATIONAL PARAMETER'
        ACCEPT*,A
        A2=A*A
        C1=3.*D*A2                      ! CONSTANTS TO EVALUATE
        C2=D*A2*A2                      ! LOCAL ENERGY
        EAV=0.                          ! TO COMPUTE AVERAGES
        EAV2=0.
        NACEP=0                         ! COUNT NUMBER OF ACCEPTS.
        X=RAN(NR)-0.5                   ! STARTING POINT
        Y=RAN(NR)-0.5
        Z=RAN(NR)-0.5
        R=X*X+Y*Y+Z*Z
        FIOLD=FI(R,A2)                  ! OLD W.F. SQUARE
        ENERG=0.
        ENERG2=0.
        DO 100 ITER=1,NITER             ! RANDOM SAMPLING LOOP
        XP=X+SCAL*(RAN(NR)-0.5)         ! ATTEMPT A MOVE
        YP=Y+SCAL*(RAN(NR)-0.5)         ! MAXIMUM OF 0.5*SCAL
        ZP=Z+SCAL*(RAN(NR)-0.5)         ! IN EACH AXIS
        RP=XP*XP+YP*YP+ZP*ZP
        FINEW=FI(RP,A2)                 ! NEW W.F. SQUARE
C       ----------------------------------------------------------
        IF(FINEW/FIOLD .LT. RAN(NR)) GOTO 5000  ! MOVE REJECTED    !
        X=XP                                    ! MOVE ACCEPTED    !
        Y=YP     !-----------------!            ! UPDATE POSITIONS !
        Z=ZP     ! METROPOLIS BLOCK !                              !
        R=RP     !-----------------!                               !
        FIOLD=FINEW                             ! AND WAVE FUNCTION!
        NACEP=NACEP+1                           ! COUNT ACCEPTS    !
5000    CONTINUE                                                   !
C       ----------------------------------------------------------
        EN=V(R)+C1-C2*R                 ! COMPUTE LOCAL ENERGY
        ENERG=ENERG+EN                  ! AND AVERAGES
        ENERG2=ENERG2+EN**2
100     CONTINUE
        E=ENERG/NITER
        E2=ENERG2/NITER
        ERR=SQRT((E2-E*E)/NITER)
        PRINT*,' ENERGY OF G.S.=',E,' ERROR=',ERR
        PRINT*,' NUMBER OF SAMPLES=',NITER, 'NUM. ACCEPTED=',NACEP
        END
```

Figure 4.2. Variational Montecarlo with Metropolis Random Walk method to generate random points. The program corresponds to the example of section 4.2.

Figure 4.1 contains a simple Fortran program corresponding to the problem at hand with a direct sampling from the squared wave function. The bulk of the program is the set of statements in DO-loop labelled 100, its structure being quite obvious. Draw a point (its three cartesian coordinates), with subroutine GAUSS, compute the local energy and accumulate to compute at the end the local energy average and the squared local energy average. The method of drawing normal distributed random points was already explined in Section 2.3. With the algorithm presented there, random numbers are generated by pairs, and the routine takes advantage of this fact by generating alternatively four points, and saving one for next time, and two points, recovering the previously saved value. This is controlled by the logical variable ODD.

Figure 4.2 is the Fortran program for the same problem but using now Metropolis method to generate the sequence of random positions. The structure of the program is very much the same than the structure of the previous program, with small differences. First of all, Metropolis algorithm consists in a move from the old position (X,Y,Z) to a new position (XP,YP,ZP), moving symmetrically around the starting position. The maximum length of the step at each move is controlled by the variable SCAL, which should be adjusted so as to have a reasonable ratio of acceptances. The block between the two COMMENT lines with equal signs is the acceptance/rejection test.

Table 4.1. Variational determination of the ground state of a gaussian potential with various methods. All energies are in MeV and the variational parameter a is in 1/fm. The first row of the table indicates the number of samples.

	Analytic		Gaussian sampling		Metropolis sampling	
a	$\langle E_L \rangle$	$\langle E_L^2 \rangle$	1000	10000	1000	10000
0.4	-6.894	120.3	-6.58±.26	-6.89±.08	-8.19±.30	-7.07±.09
0.5	-9.190	135.6	-9.07±.22	-9.11±.07	-9.61±.23	-9.12±.07
0.6	-10.301	131.2	-10.49±.16	-10.31±.05	-10.86±.16	-10.45±.05
0.7	-10.069	121.5	-9.94±.13	-10.10±.04	-9.66±.10	-9.98±.04
0.8	-8.525	130.2	-8.70±.24	-8.53±.08	-9.13±.24	-8.45±.08

Some results corresponding to this case example are presented in Table 4.1. The columns labelled ANALYTIC show the exact values, and include both the energy expectation value and the expectation value of the hamiltonian squared. The next two columns are the results corresponding to a direct generation of random numbers from the gaussian and the last two columns are the results corresponding to Metropolis method. The calculations have been carried out with the programs listed above, for 1000 and 10000 samples, and the scale factor was 3 fm, which gives around a 50% of acceptances.

The columns labelled GAUSSIAN are in a very good agreement with the analytic results. In all cases the obtained values are within one standard deviation of the exact values. This is not the case of

Metropolis method with 1000 points, whereas the calculation with 10000 is again in very good agreement with the exact values. The conclusion is that 1000 steps are not enough to randomly cover all configuration space and that the random walk must be longer. In this case the value of the statistical error is meaningless. In real problems the excat value is not known a priori, and a large amount of computing is necessary to ascertain the validity of Montecarlo estimates.

4.3 Extended systems and periodic boundary conditions

Certainly one cannot attempt the direct description of a system with an infinite number of particles. The usual way of dealing with this problem, /7/, /11-12/, inherited form Classical Statistical Mechanics, is assuming that the system is partitioned in an infinite set of simulation boxes of side L which are repetitions of a fundamental cell. This picture is known with the name of establishing periodic boundary conditions.

The number of particles in each cell and the length of the simulation cube L are chosen so as to have the desired density

$$\rho = \text{number of particles} / L^3 \qquad (4.15)$$

The value of the number of particles is taken so as to allow for a crystalline order in the case this is the phase preferred by the system. At high density one can expect a close packing of the atoms, and the expected structure will be fcc (face centered cubic) or hcp (hexagonal close packed). In a fcc structure the number of atoms per unit cell is four, 6 atoms at the corners each shared by six cells plus six atoms at the faces each shared by two cells. If the simulation cube contains n^3 cells, the total number of particles will be $4n^3$. The 'magic numbers' generated in this form are 32, 108, 256, 500, 864, ...and most of variational Montecarlo work in bosons is carried out with one of these numbers.

The magic numbers corresponding to fermion systems have a different origin, being related to the quantization of single particle states momentum, and the occupancy of levels up to a given value of the Fermi momentum. The allowed values of the momenta are determined in terms of three integer numbers (n_1, n_2, n_3), positive or negative, and levels are filled in increasing order of the distance to the origin. The first point is (0,0,0) which corresponds to 1 state. At a distance of 1 we may have (1,0,0), (0,1,0), and (0,0,1), and the corresponding negative values, with a total of 6 new states. Next, at distance $\sqrt{2}$ we have (1,1,0), (1,0,1) and (0,1,1) and the negative values, with a total of 12 states, and so on. The cumulative numbers are 1,7,19,27,33,57,... to be multiplied times the degeneracy (2 or 4) of the system.

To use consistently periodic boundary conditions is necessary to follow certain rules. Each time a particle exits of the fundamental cell (i.e., the attempted and succesful Metropolis move forces the particle to go outside the simulation cube) another image of this particle will enter through the opposite wall. To test the acceptance of a possible move, one should test the change in the wave function due to move of this particle with consideration of all boxes. To make the computation feasible, it is usually assumed that the correlation factor is unity at distances greater than L/2. Then, we draw a sphere of radius L/2 around the particle under consideration, and count the contribution to the wave function of all particles inside this sphere, even if some of them are not in the fundamental cell. If this sphere contains a given particle, then none of the other periodic images of this particle will be inside the sphere. In conclusion, to compute the change of the wave function due to the motion of one particle, we must determine only its nearest neighbours generated from the particles in the fundamental cell by adding and substracting the length L to any of its cartesian coordinates. This mechanism is depicted in Figure 4.3

Once all particles have been tested by means of Metropolis algorithm, we have to compute the local energy corresponding to this new configuration. The local energy has two terms, one related to the interaction potential and the other involving derivatives of the correlation factor. Because of the healing conditions of the correlation factor at L/2, only particles inside this sphere must be considered to evaluate the kinetic energy. Warning: you must ensure that the correlation factor is smooth around the distance L/2, both function and derivative, to compute correctly the kinetic energy. A simple way to overcome this question is to take a length L large enough so that the correlation is negligible at distances longer than L/2. The price of this solution is that a large number of particles is necessary at high densities.

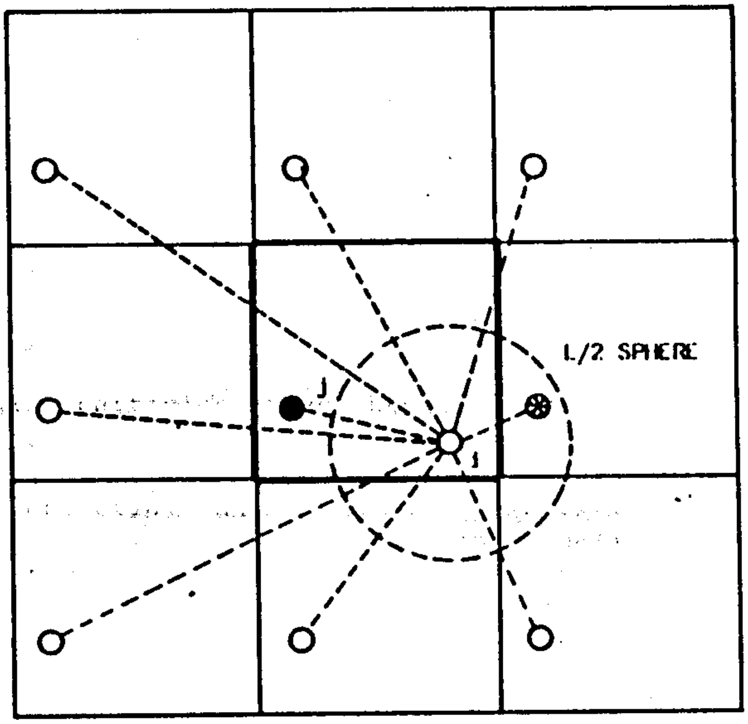


Figure 4.3. Determination of the nearest neighbour of particle i related to particle j (black circle) and all its images (open circles). The circle with an asterisk is the nearest neighbour. Note that there is only one particle of family j in the L/2 sphere drawn around particle i. This figure is adapted from Wood and Parker, /11/.

All these problems are not present when computing potential energy. In that case we should consider the interaction of particle i with particle j and all images of particle j, and this for all pairs of particles of the fundamental cell. If the interaction is short ranged we can discard all particles at a distance larger than an appropriate cutoff value. However, in the case of long range interactions, like e.g. Coulomb potential, all particles must be considered. The way of doing it is to compute Ewald's like summations.

The effect of these cutoffs related to boundary conditions can be corrected in the case of simple interactions. This will be explicitly discussed in the next section for the case of a Lennard-Jones interaction.

4.4. Study of liquid Helium 4

Liquid helium 4 was the first quantum system studied with variational Montecarlo method, already in 1965 by McMillan /13/. The interaction between helium atoms is quite well described in terms of a 6-12 Lennard-Jones potential

$$V(r) = 4\varepsilon\left(\left\{\frac{\sigma}{r}\right\}^{12} - \left\{\frac{\sigma}{r}\right\}^{6}\right) \qquad (4.16)$$

where σ and ε are parameters characteristic of helium adjusted by De Boer and Michels to the values $\sigma=2.556$ A and $\varepsilon=10.22$ K. The other constant describing the system is the helium mass, its value being $\hbar^2/m=12.1209$ KA^2. A simple form of the two body correlation factor is

$$f(r_{ij}) = \exp\left(-\frac{1}{2}\left\{\frac{b}{r}\right\}^{n}\right) \qquad (4.17)$$

This correlation factor satisfies the Schrodinger equation for the relative motion of two particles at short distances, i.e. in the region of repulsion, when n=5

$$\left[-\frac{h^2}{m}\nabla^2 + V(r)\right] f(r) = 0 \qquad (4.18)$$

actually, only the dominant term of this equation r^{-12} is satisfied. The corresponding value of b is

$$b = (\,16\varepsilon\, m/25\,\hbar^2\,)^{1/12}.\sigma \qquad (4.19)$$

corresponding to 1.13 sigmas. With this simple form of the correlation the only free parameters are the correlation range b, the power n and the density of the system. This form of the correlation factor was suggested by McMillan /13/, and subsequently used by Schiff and Verlet in a molecurar dynamics study of the quantum problem. Attempts to improve this two body correlation have resulted in a very small gain in energy (see the review article of Ceperley and Kalos /15/ for a complete analysis).. We should however mention that an important correction to the simple form (17) is the introduction of three body correlations /16/, particularly at high densities.

Table 4.2 shows some of the helium calculations using Lennard-Jones interaction, with variational Montecarlo and two body correlations only and including three body correlations, and also with the exact Green functions Montecarlo method, to be studied in chap. 7. From this table we observe that McMillan's form explains almost all binding energy. The minimum is obtained at a density 0.9 times the experimental value, and at n=5 and b=1.16 sigmas, very close to the suspected value. Three body correlations are also important giving a value for the energy very close to the exact GFMC method. Note finally that GFMC energies are still a five per cent higher (less bound) than the experimental value. To obtain the experimental value is then necessary to improve the two body interaction. Nowadays it is beleived that Aziz interaction /17/ is the best interaction describing helium liquids.

Table 4.2: Liquid helium 4 with Lennard Jones potential

	Two body correlation	Three body correlation	GFMC	Experiment
E(K)	-5.9 ± 0.1	-6.53 ± 0.02	-6.84 ± 0.02	-7.14
Density (σ^{-3})	0.333	0.365	0.365	0.3648

In the rest of this section I will describe the simplest variational calculation using the correlation (17) with n=5. The squared wave function is given by

$$\Psi^2 = \exp\left\{ - \sum_{i<j} \left[\frac{c}{x_{ij}}\right]^5 \right\} \tag{4.20}$$

where $c=b/\sigma$, and $x_{ij}=r_{ij}/\sigma$ are the dimensionless interatomic distances.The local energy can be reduced by means of an integration by parts to the two body form /14/

$$E = \sum_{i<j} \left\{ \frac{5\hbar^2 c^5}{m\sigma^2} \; \frac{1}{x_{ij}^7} + 4\varepsilon \left[\frac{1}{x_{ij}^{12}} - \frac{1}{x_{ij}^6} \right] \right\} \tag{4.21}$$

The best way to describe all computing steps is to follow the Fortran code shown in Figure (4.4). Note that this code has not been optimized for the sake of simplicity. However, it works and produces correct results.

The positions of the particles are stored in array X(I,J), the first index corresponding to the component (I=1,2,3) and the second to the particle. DO loop 10 places all particles at random inside the simulation cube and the purpose of DO loop 100 is to carry out a series of statistically independent averages to evaluate the mean value of the energy and its statistical error. Each of these calculation blocks is carried out in DO loop 200, which is a long series of sequential moves of all particles. Particles are moved inside a cube of side SCAL/2, the value of SCAL being fixed in the program to one fourth of the simulation cube side L. In the range of b values of interest this results in around a 50% of acceptances. Of course, the value of SCAL may be changed to improve the stability or just to understand its actual role in the calculation.

Moves are carried out in DO loop 1000. From the old position we attempt a new position in the usual way. Afterwards, the new coordinates are displaced, if necessary, to replace particles in the fundamental cell. Once a new position is determined, we must compute the quotient of the old wave function by the new wave function by means of eq. (20). However, given that only one particle has been moved, only the terms in (20) involving this particle will contribute to the change of the wave function and it is not necessary to compute all pairs of particles in eq. (20). This part of the calculation is made in DO loop 320, running over all particles with the exception of the particle we are attempting to move. Each of the contributions to the exponential part of eq. (20) requires the determination of the nearest neighbour (this is done in subroutine CLOSE) and also the test on the cutoff of the correlation factor, i.e., if the distance to the nearest neighbour is greater than L/2 it is assumed to be L/2, so that the correlation is constant outside the interesting region.

The rest of the code is as expected: Metropolis test is carried out, and once all particles have been tested the local energy is computed and accumulated for the statistical average.

The determination of the nearest neighbour is the task of subroutine CLOSE, and given that the algorithm is quite straightforward I beleive it does not require further explanations. This routine also determines the cartesian coordinates of the nearest neighbour with respect to the particle under consideration. Conventionally these coordinates have been taken as positive always (even if it is not the case) and their value is required by the routine ENER which computes the local energy.

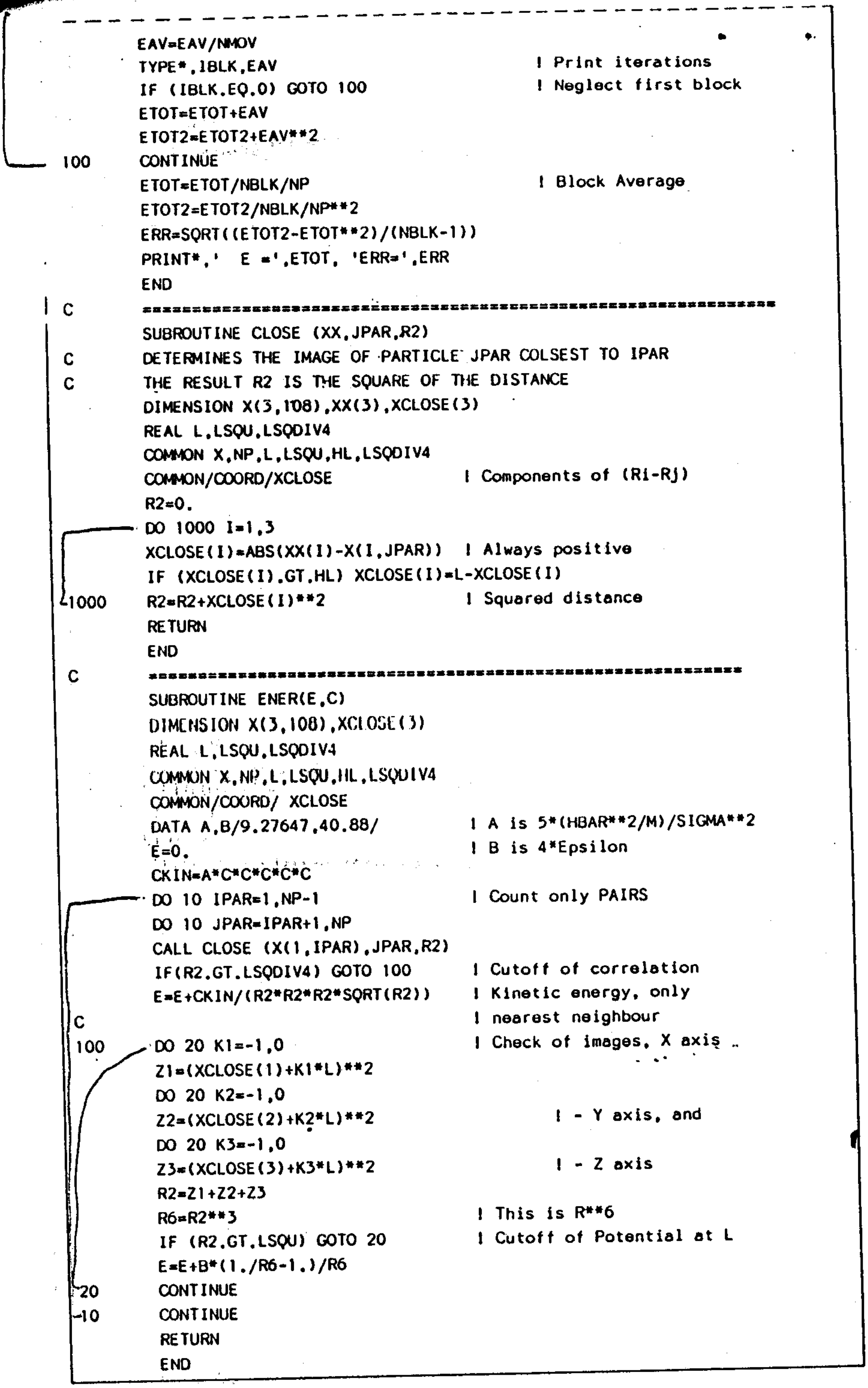
In the calculation of the local energy, the kinetic part has a cutoff at L/2, as required by consistency with Metropolis algorithm. However, the potential energy has been computed with a larger cutoff (L), and because of that more than one of the images can contribute to it. This is taken into account in Do loops 20. Here is where our criterium of considering all cartesian coordinates positive enters: the only images of interest are obtained with no displacement or with a -L displacement from the nearest neighbour.

Because of the cutoff in the interaction, the final energy must be corrected to have the physical value. A simple way of evaluating this correction is the following /14/. If the cutoff distance is large enough, we may consider that the two body distribution function is already 1 at distances longer than the cutoff. The contribution to the energy per particle coming from large distances can be approximated by

$$\Delta E = \frac{1}{2} \int_L d^3r \; g(r) \; V(r) = \frac{1}{2} \int_L d^3r \; V(r) \qquad (4.22)$$

See the lectures of Sergio Rosati to find the definition of the two

```
C       MACMILLAN'S VARIATIONAL MONTECARLO IN LIQUID HELIUM-4
C       WITH LENNARD-JONES POTENTIAL
        DIMENSION X(3,108),XOLD(3),XNEW(3)
        REAL L,LSQU,LSQDIV4
        COMMON X,NP,L,LSQU,HL,LSQDIV4     ! Arguments for subroutines
        TYPE*,' INTEGER FOR RANDOMNESS, NUMBER PARTICLES AND CORRELATION'
        ACCEPT*,NR,NP,C                   ! C=B/SIGMA, Dimensionless
        DENS=0.3648*~~0.0~~               ! SCHIFF-VERLET in SIGMA units
        L=(NP/DENS)**0.3333333            ! Simulation cube
        LSQU=L*L
        LSQDIV4=LSQU/4.
        HL=L/2.
        SCAL=L/4.                         ! Scale of Metropolis moves
        C5=C*C*C*C*C
        TYPE*,' ENTER NUMBER BLOCKS and NUMBER OF MOVES'
        ACCEPT*,NBLK,NMOV
        DO 10 I=1,NP                      ! Starting position
        DO 10 J=1,3                       ! at random in simulation box
        X(J,I)=L*RAN(NR)
10      CONTINUE
C       METROPOLIS MOVES
        ETOT=0.                           ! Block average
        ETOT2=0.
        DO 100 IBLK=0,NBLK
        EAV=0.
        DO 200 IMOV=~~1~~,NMOV
        DO 300 IPAR=1,NP                  ! Move particles sequentially
        DO 1000 I=1,3
        XOLD(I)=X(I,IPAR)                          ! Old position
        XNEW(I)=XOLD(I)+SCAL*(RAN(NR)-0.5)         ! Attempted move
        IF(XNEW(I).LT.0.) XNEW(I)=XNEW(I)+L        ! Move particle to the
        IF(XNEW(I).GT.L)  XNEW(I)=XNEW(I)-L        ! fundamental cell
1000    CONTINUE
        SUM=0.                                     ! Old and New W.F.
        DO 320 JPAR=1,NP
        IF (JPAR.EQ.IPAR) GOTO 320                 ! Skip particle IPAR
        CALL CLOSE (XOLD,JPAR,R2)                  ! Old neighbour: R2=squared
        IF (R2.GT.LSQDIV4) R2=LSQDIV4              ! Cuttoff at L/2
        CALL CLOSE (XNEW,JPAR,S2)                  ! New neighbour: S2=squared
        IF (S2.GT.LSQDIV4) S2=LSQDIV4              ! and cuttoff
        SUM=SUM+1./(R2*R2*SQRT(R2)) - 1./(S2*S2*SQRT(S2))
320     CONTINUE
C       QUOTIENT OF PROB: (W.F.NEW/W.F.OLD)**2
        EXPR=C5*SUM                                ! Exponent
        IF(EXPR.GT.0.) GOTO 330                    ! Accepted
        PROBQ=EXP(C5*SUM)                          ! Probability quotient
        IF(PROBQ.LT.RAN(NR)) GOTO 300              ! Test Acceptance-Rejection
330     DO 1010 I=1,3                              ! Accepted move
1010    X(I,IPAR)=XNEW(I)                          ! Update position
300     CONTINUE
        CALL ENER(E,C)                             ! Sample energy
        EAV=EAV+E
200     CONTINUE
```



```
      EAV=EAV/NMOV
      TYPE*,IBLK,EAV                        ! Print iterations
      IF (IBLK.EQ.0) GOTO 100               ! Neglect first block
      ETOT=ETOT+EAV
      ETOT2=ETOT2+EAV**2
100   CONTINUE
      ETOT=ETOT/NBLK/NP                     ! Block Average
      ETOT2=ETOT2/NBLK/NP**2
      ERR=SQRT((ETOT2-ETOT**2)/(NBLK-1))
      PRINT*,'  E =',ETOT, 'ERR=',ERR
      END
C     ==============================================================
      SUBROUTINE CLOSE (XX,JPAR,R2)
C     DETERMINES THE IMAGE OF PARTICLE JPAR COLSEST TO IPAR
C     THE RESULT R2 IS THE SQUARE OF THE DISTANCE
      DIMENSION X(3,108),XX(3),XCLOSE(3)
      REAL L,LSQU,LSQDIV4
      COMMON X,NP,L,LSQU,HL,LSQDIV4
      COMMON/COORD/XCLOSE                ! Components of (Ri-Rj)
      R2=0.
      DO 1000 I=1,3
      XCLOSE(I)=ABS(XX(I)-X(I,JPAR))     ! Always positive
      IF (XCLOSE(I).GT.HL) XCLOSE(I)=L-XCLOSE(I)
1000  R2=R2+XCLOSE(I)**2                 ! Squared distance
      RETURN
      END
C     ==========================================================
      SUBROUTINE ENER(E,C)
      DIMENSION X(3,108),XCLOSE(3)
      REAL L,LSQU,LSQDIV4
      COMMON X,NP,L,LSQU,HL,LSQDIV4
      COMMON/COORD/ XCLOSE
      DATA A,B/9.27647,40.88/            ! A is 5*(HBAR**2/M)/SIGMA**2
      E=0.                               ! B is 4*Epsilon
      CKIN=A*C*C*C*C*C
      DO 10 IPAR=1,NP-1                  ! Count only PAIRS
      DO 10 JPAR=IPAR+1,NP
      CALL CLOSE (X(1,IPAR),JPAR,R2)
      IF(R2.GT.LSQDIV4) GOTO 100         ! Cutoff of correlation
      E=E+CKIN/(R2*R2*R2*SQRT(R2))       ! Kinetic energy, only
C                                        ! nearest neighbour
100   DO 20 K1=-1,0                      ! Check of images, X axis
      Z1=(XCLOSE(1)+K1*L)**2
      DO 20 K2=-1,0
      Z2=(XCLOSE(2)+K2*L)**2                     ! - Y axis, and
      DO 20 K3=-1,0
      Z3=(XCLOSE(3)+K3*L)**2                     ! - Z axis
      R2=Z1+Z2+Z3
      R6=R2**3                           ! This is R**6
      IF (R2.GT.LSQU) GOTO 20            ! Cutoff of Potential at L
      E=E+B*(1./R6-1.)/R6
20    CONTINUE
10    CONTINUE
      RETURN
      END
```

Figure 4.4. Variational calculation of liquid He-4 with Lennard-Jones interaction.

body distribution function and eq. (22).

The integration in (22) is extended to all space outside the sphere of radius L. The most important contribution to eq. (22) comes from the term r^{-6} in Lennard-Jones interaction, and the value of the energy correction which results is

$$E = - \frac{2\pi}{3} \; 4 \varepsilon \rho \sigma^6 / L^3 \qquad (4.23)$$

In the case of 108 particles at a density 0.9 times the equilibrium density the value of this correction is -0.08K.

4.5 Fermion systems

There are not essential differences in the case of fermion systems with respect to the case of boson systems. There are however some technical problems whose main effect is the increasing of computing time. The first problem is the mere calculation of determinants at every Metropolis move. The second problem is the need of taking into account spin and isospin degrees of freedom in the case the interaction or the correlation or both depend on such operators.

The trial wave function is now the product of the Jastrow correlation and a Slater determinant. In the case of spin independent interactions there are as many Slater determinants as fermion species, e.g. one corresponding to spin up and the other corresponding to spin down. Note that one can include also various kinds of two body correlations, namely, spin up-spin up, up-down and down-down correlations. Each determinant is constructed with single particle orbitals φ_α , so that the fermion part is $\det(D_{i\alpha})$, with $D_{i\alpha} = \varphi_\alpha(r_i)$. Greek indexes will run over orbitals, and roman indexes over particles. Once the position of the particles is determined, this determinant is a number and the acceptance criterium in a Metropolis move is the squared quotient of two determinants, apart from the correlation factor.

Here is where the problem comes in, because the number of floating operations involved in the computation of a determinant is proportional to N^2 and consequently moving N particles involves N^3 operations. Ceperley, Chester and Kalos /19/ have found an algorithm to reduce the number of operations to N^2. The clue is to use and update the matrix of cofactors $C_{i\alpha}$. Each element of this matrix is obtained by zeroing in the original matrix row i and column α , but leaving the element (i,α) equal to 1. Matrix C is related to the inverse traspose of matrix D,

$$C_{i\alpha} = \det(D) \, \overline{D}_{i\alpha} \qquad (4.24)$$

Once C is known, the value of the determinant may be computed as

$$\det(D) = \sum_\alpha C_{i\alpha} \; D_{i\alpha} \qquad (4.25)$$

where the sum runs only over α , the particle label i being arbitrary.

→ desarrollo por adjuntos de un determinante!

This arbitrarieness is the clue of the simplicity of the subsequent algorithm.

Let us attemp a move of particle i. Then:

i. Matrix D does not change, with the exception of row i; its new value being

$$D'_{i\alpha} = \varphi_\alpha(r'_i) \quad , \alpha = 1, .. N \qquad (4.26)$$

ii. All elements of matrix C will change, with the exception of row i which remains with its old value.

The new determinant is

$$\det(D') = \sum_\alpha C_{i\alpha} \; \varphi_\alpha(r'_i) \qquad (4.27)$$

where i corresponding to the particle moved. The ratio of the new to the old determinant is (put eqs. (24) and (27) together)

$$q = \frac{\det(D')}{\det(D)} = \sum_\alpha \overline{D}_{i\alpha} \; \varphi_\alpha(r'_i) \qquad (4.28)$$

This part of the calculation is simple, and involves only N floating operations. If the move is accepted, all of the elements of D must be updated. The way of doing it is the following:

$$\overline{D}_{i\alpha} \longrightarrow \overline{D}_{i\alpha} / q \qquad , \alpha = 1, ... N$$

$$\overline{D}_{j\alpha} \longrightarrow \overline{D}_{j\alpha} - \frac{\overline{D}_{i\alpha}}{q} \sum_\beta \overline{D}_{j\beta} \; \varphi_\beta(r'_i) \quad , \alpha = 1 ... N, \text{ and } j=i \quad j \neq i \; ? \qquad (4.29)$$

which involves N^2 operations. Ceperley and coworkers /19/ discuss also the way of computing the energy from matrix D.

The question of spin dependence is quite more involved. The way of dealing with this problem is to carry out a double random walk to integrate over the spatial coordinates and to sum up over spin and isospin variables.

Assume an ordering convention so that a given single particle state $|\phi_\alpha\rangle$ is the direct product of a coordinate space state $|\varphi_\alpha\rangle$ and a spin state $|\chi_\alpha\rangle$.

$$|\phi_\alpha\rangle = |\varphi_\alpha\rangle \, |\chi_\alpha\rangle \qquad (4.30)$$

The numerical value, i.e., the wave function, is obtained by projecting on the bra $\langle r\,\nu|$, where ν is a discrete index representing the two spin orientations or the four spin/isospin orientations. Then

$$\phi_\alpha(r,\nu) = \langle r\,\nu \,|\, \varphi_\alpha \chi_\alpha\rangle \qquad (4.31)$$

and its value is $\varphi_\alpha(r)$ if χ_α is state ν, and 0 otherwise.

This is extended to a many fermion state, so that, e.g.,

$$\langle\Psi|\Psi\rangle = \sum_{\nu_1 \ldots \nu_N} \int dr_1 \ldots dr_N \langle\Psi|r_1\nu_1, \ldots\rangle\langle r_1\nu_1, \ldots|\Psi\rangle \qquad (4.32)$$

Note that $|\Psi\rangle$ is antisymmetric under permutations of single particle labels, but the intermediate states are not necessarily restricted. Of course, non antisymmetric intermediate states will give a null contribution to (32).

The numerical value of the wave function is a Slater determinant with many zeros. For example, assume a spin 1/2 four particle system with two spatial orbitals φ_1 and φ_2. The single particle states may be named in the following form

$$\begin{aligned} |1\rangle &= |\varphi_1, +\rangle \\ |2\rangle &= |\varphi_1, -\rangle \\ |3\rangle &= |\varphi_2, +\rangle \\ |4\rangle &= |\varphi_2, -\rangle \end{aligned} \qquad (4.33)$$

We may construct $2^4=16$ Slater determinants corresponding to the possible values of ν_i. Only that cases in which two particles have spin up and two particles have spin down are not null. This is the case of

$$\langle r_1+, r_2+, r_3-, r_4-|\Psi\rangle = \begin{vmatrix} \varphi_1(r_1) & \varphi_1(r_2) & 0 & 0 \\ 0 & 0 & \varphi_1(r_3) & \varphi_1(r_4) \\ \varphi_2(r_1) & \varphi_2(r_2) & 0 & 0 \\ 0 & 0 & \varphi_2(r_3) & \varphi_2(r_4) \end{vmatrix} \qquad (4.34)$$

A Metropolis move is carried out in two steps:

i. Try a displacement of particle i,

$$r_i \longrightarrow r_i + d(\xi \ -0.5)$$

ii. Try a swap of spin of particle i with the spin of any of the particles, itself included.

The first part results in changing the value of the coordinate r_i. The second part acts like a column exchange in the determinant. For example, interchanging the spins of particles 1 and 3 in determinant (34) gives a new determinant

$$
\langle r_1 -, \ r_2 +, \ r_3 +, \ r_4 - | \Psi \rangle = \begin{vmatrix} 0 & \varphi_1(r_2) & \varphi_1(r_3) & 0 \\ \varphi_1(r_1) & 0 & 0 & \varphi_1(r_4) \\ 0 & {}_2(r_2) & {}_2(r_3) & 0 \\ \varphi_2(r_1) & 0 & 0 & \varphi_2(r_4) \end{vmatrix} \qquad (4.35)
$$

Note that in all cases the determinant degenerates in two blocks after an appropriate reordering of rows and columns.

Finally, the energy is sampled with the hamiltonian matrix

$$
\langle \ |H| \ \rangle = \int dr_1 dr_2 \ldots dr_N \sum_{\nu_1 \cdots \nu_N} \sum_{\mu_1 \cdots \mu_N} \langle \Psi | r_1 \nu_1 \ \ldots \ r_N \nu_N \rangle \qquad (4.36)
$$

$$
\langle r_1 \nu_1 \ \ldots \ r_N \nu_N | H | r_1 \mu_1 \ \ldots r_N \mu_N \rangle \langle r_1 \mu_1 \ \ldots r_N \mu_N | \Psi \rangle \ .
$$

To complement this brief section on fermion problems we suggest the reader to consult Refs. /20/ for the case of few particles, and Ref. /21/ for a case with more particles.

4.6 Variational Montecarlo: summary

We have seen that Metropolis algorithm can be used to sample from almost everything. In other words, from the variational point of view almost any kind of trial wave function describing a many body system can be used. Note however that programming codes must be carefully writen to make calculations feasible.

Many systems of bosons and fermions have been studied in this form. Generally the calculations have been limited to infinite systems with spin independent interactions. A good description of many of the physically interesting systems studied with VMC can be found in the review article of Ceperley and Kalos /15/. We should say, however, that this algorithm (as any other many body theory) has many troubles when dealing with spin or isospin dependent interactions.

A final question, and not trivial, is that we are interested in the minimum value of the energy. This means that each VMC calculation must be repeated several times for different values of the parameters describing the trial function. This is a quite slow method implying a huge amount of computing time consumption. The problem is still a little more complex because the statistical fluctuations of the results may give rise to an incorrect placement of the minimum. A procedure appropriate to simple forms of the interaction was discovered by McMillan /13,14/. His method, based in scaling properties of the various terms of the local energy, seems not to be known in other many body theories. Another nice and general purpose method, known as the method of reweighting configurations, is described in Ref. 19/.

5. MONTECARLO INTEGRATION OF THE MANY BODY SCHRODINGER EQUATION

A completely different approach to the many body problem in the framework of Montecarlo is the direct integration of Schrodinger equation. Two methods have been developed within this spirit, which are known as DIFFUSION MONTECARLO (DFMC) and GREEN FUNCTION MONTECARLO (GFMC). The first method integrates the imaginary time dependent Schrodinger equation, whereas the second integrates directly the time independent Schrodinger equation. Actually, GFMC is a time-integrated version of DFMC.

These two methods have many features in common. Both represent the many body wave function by means of a large set of multidimensional points R_i, where as usual R represents the set of 3N coordinates of a N particle system. This set is such that its elements are distributed accordingly with the ground state wave function of the system. Each point R is called a psip in DFMC or a walker in GFMC. It is very important to stress that the quantity interpreted as a probability distribution function is the wave function of the system, and not its square, as in VMC. Of course, this requires the wave function to be positive, as in the case of a boson ground state of a given hamiltonian.

This very particular way of representing the wave function is depicted in Figure (5.1) for a one dimensional case. The set of random points, which are vertical bars along the X-axis, is more concentrated in the regions where the wave function is important, and their density is low in the regions where the wave function is small. Along the iterations, going from a wave function to another wave function means to change the random numbers of the set (called a generation) to a new set. Each psip or walker generates a number of descendants (none, one or more than one) which will accomodate to the new probability distribution function.

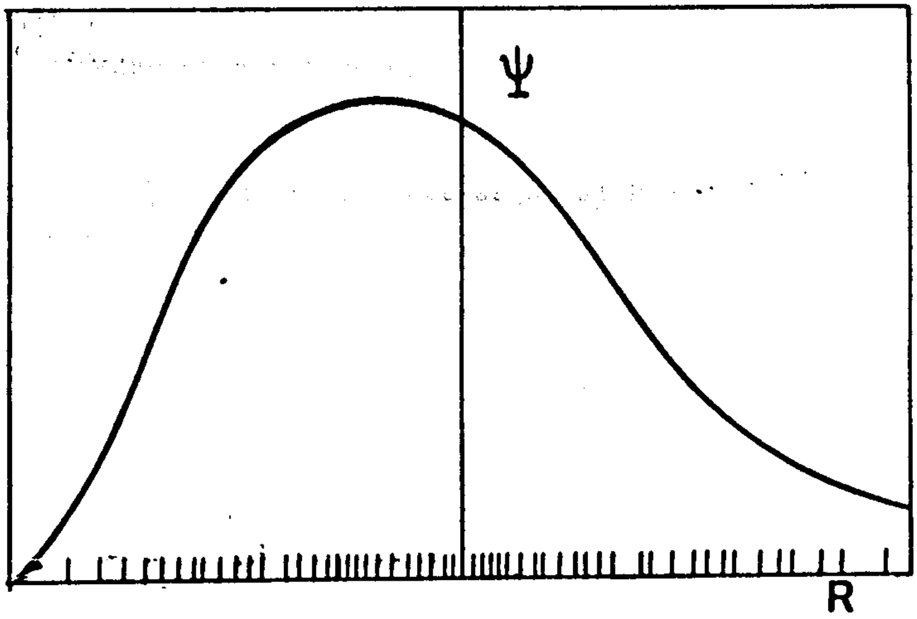


Figure 5.1 Montecarlo interpretation of the wave function

Both methods are iterative, and the ground state is approached exponentially in both methods. Again the technique of importance sampling may be introduced, improving the quality of the results as well as reducing computer time. The way of taking into account importance sampling is quite similar in both theories.

In their simplest form, DFMC and GFMC are only appropriate to describe the ground state of boson systems. In the case of fermions the ground state wave function is not everywhere positive, so the wave function cannot be interpreted as a probability distribution function. Importance sampling combined with the so called fixed node approximation permits the description of fermion systems.

To end this short introductory chapter let me mention that both techniques are intrinsically exact, with only statistical errors which may be controlled by increasing the statistics or by improving the importance sampling function. In the case of DFMC there is an additional error coming from the step by step integration in time.

I will discuss first DFMC method in a rather detailed form, in chapter 6. Finally chapter 7 will be devoted to GFMC.

6. DIFFUSION MONTECARLO

6.1 Introduction

The possibility of integrating the time dependent Schrodinger equation by means of stochastic methods was suggested by Metropolis and Ulam many years ago /22/, and subsequently applied by Donskar and Kac /23/. Undoubtedly, computers at that time were small and slow, and the idea remained dormant until 1975, when Anderson /24/ applied it to an atomic system. Since that time the method has been deeply developed /25-33/ and many applications have appeared in the mean time, mainly in the field of quantum chemistry and electron systems. Just for didactical interest I suggest the introductory reader to study the works of Reynolds et al /30/ and Vrbik and Rothstein /33/.

The starting point is the Schrodinger equation in imaginary time,

$$-\frac{\partial\Psi(R,t)}{\partial t} = (H - E)\,\Psi(R,t) \qquad (6.1)$$

where R represents the set of all coordinates and E is an energy shift, at this moment arbitrary. In eq. (1) time is measured in units of inverse energy.

The formal solution of eq. (1) is

$$\Psi(t) = \exp\left\{-t(H-E)\right\}\Psi(0) \qquad . \quad (6.2)$$

We may expand the starting function in the basis of eigenfunctions of H,

$$H\ \phi_n = E_n\ \phi_n \qquad E_0 < E_1 < \dots \tag{6.3}$$

which have been ordered by their energy. Then eq. (2) is converted into

$$\psi(0) = \sum C_n\ \phi_n \qquad \psi(t) = \sum C_n\ \phi_n\ \exp\{-t(E_n - E)\} \tag{6.4}$$

The amplitudes change with time, increasing or decreasing accordingly with the sign of (E_n-E). Independently of the value of the energy shift E, the most important amplitude at large time is that one corresponding to the state with energy E_0, i.e., the ground state, provided the starting wave function has a non null projection of the g.s., i.e., provided C_0 is not null.

This behaviour may be interpreted in terms of a diffusion problem. With the usual form of the hamiltonian, eq. (1) is

$$-\frac{\partial \psi(R,t)}{\partial t} = -D\nabla^2\psi(R,t) + V(R)\ \psi(R,t) - E\ \psi(R,t) \tag{6.5}$$

where $D=\hbar^2/2m$ is called the diffusion constant, and ∇^2 is a shorthand representation of the Laplace operator on all coordinates. This term represents an isotropic diffusion. The other term, (V(R)-E) is a rate term, with creation or dissapearance of the magnitude diffused. These two concepts will be clarified below.

Equation (5) may be improved in the sense of importance sampling introduced at the beginning of these lectures. Consider a trial function $\psi_T(R)$, independent of time, and construct an equation analogous to (5) for the quantity

$$f(R,t) = \psi_T(R)\ \psi(R,t) \tag{6.6}$$

This equation turns out to be

$$-\frac{\partial f(R,t)}{\partial t} = -D\nabla^2 f(R,t) + D\vec{\nabla}(\vec{F}\ f(R,t)\ ) + (E_L(R)-E)\ f(R,t) \tag{6.7}$$

where F is called the drift force, and given by

$$\vec{F} = \frac{2}{\psi_T(R)}\vec{\nabla}\ \psi_T(R)\ , \tag{6.8}$$

and is directly related to the trial wave function. E_L is the local energy, the same quantity introduced in VMC,

$$E_L\ (R) = \frac{1}{\psi_T(R)}\ \ H\ \psi_T(R) \tag{6.9}$$

The objective of DFMC is the integration in time, step by step, of eq. (5) or eq. (7), up to a value of the time large enough so that the only important amplitude is the corresponding to the ground state wave function. This integration is carried out by means of the integral equation equivalent to (5) or (7), by using a Montecarlo description of the instantaneous wave function through a set of random points, as we have described at the end of the previous chapter.

6.2 Time dependent Green functions

The formal solution of the generic equation

$$- \partial_t \Psi = A \Psi \quad .$$

is, as we have already mentioned

$$\Psi(t) = \exp(-At) \quad \Psi(0)$$

In coordinate representation, i.e., projecting on bra $\langle R'|$, we have

$$\langle R' | \Psi(t) \rangle = \int \langle R' | \exp(-At) | R \rangle \langle R | \Psi(0) \rangle \, dR \qquad (6.10)$$

The matrix element of the exponential of A is the time dependent Green function

$$G(R',R,t) = \langle R' | \exp(-At) | R \rangle \qquad (6.11)$$

and the integral equation is usually writen in the form

$$\Psi(R',t) = \int G(R',R,t) \quad \Psi(R,0) \, dR \qquad (6.12)$$

The point is that in general G(R',R,t) is not known, but reliable approximations can be derived for small values of t. Then instead of solving directly eq. (12) one proceeds step-by-step in time, using repeteadly the formula

$$\Psi(R',t+\Delta t) = \int G(R',R,\Delta t) \quad \Psi(R,t) \, dR \qquad (6.13)$$

The next point is the calculation of the Green function for small time intervals. Let us introduce three operators related to eq. (7)

$$\begin{aligned} A_1 &= -D \nabla^2 \\ A_2 &= D \left\{ (\vec{\nabla} \vec{F}) + \vec{F} \vec{\nabla} \right\} \\ A_3 &= E_L(R) - E \end{aligned} \qquad (6.14)$$

so that A in eq. (11) is the sum $A_1+A_2+A_3$. Analogously we may define the individual Green functions

$$G_i(t) = \exp(-A_i \; t) \qquad (6.15)$$

The total Green function G may be approximated by

$$\exp(-At) = \exp(-A_3 t/2)\ \exp(-A_2 t/2)\ \exp(-A_1)$$
$$\exp(-A_2 t/2)\ \exp(-A_3 t/2) + O(t^3) \tag{6.16}$$

This may be easily checked by expanding both sides of eq. (16) in powers of t. Warning, the various operators A_i do not commute among themselves. In coordinate representation eq. (16) is converted into

$$G(R',R,t) = \int dR_1\ dR_2\ dR_3\ dR_4\ G_3(R',R_1,t/2)\ G_2(R_1,R_2,t/2)$$

$$G_1(R_2,R_3,t)\ G_2(R_3,R_4,t/2)\ G_3(R_4,R,t/2) \tag{6.16.a}$$

Eq. (16) is not the unique $O(t^3)$ approximation, but it is the most convenient form for the subsequent reduction of the total Green function.

The computation of the three individual Green functions may be carried out by integrating the differential equation they satisfy,

$$-\partial_t\ G_i(R',R,t) = A_i\ G_i(R',R,t) \tag{6.17}$$

with the boundary condition

$$G_i\ (R',R,0) = \delta(R'-R) \tag{6.18}$$

The following results are obtained

$$G_1(R',R,t) = (4\pi Dt)^{-3N/2}\ \exp\left\{-(R-R')^2/4Dt\right\}, \tag{6.19}$$

where N is the number of particles,

$$G_2(R',R,t) = \quad (R' - \mathcal{R}(t)) \tag{6.20}$$

where $\mathcal{R}(t)$ is the solution of the equation of motion

$$\frac{d\mathcal{R}}{dt} = D\ F\ (\mathcal{R}(t)) \tag{6.21}$$

$$\mathcal{R}(0) = R \tag{6.22}$$

Note that in eq. (21) $F\ (\mathcal{R}(t))$ means the drift force, eq. (8), computed at the value $\mathcal{R}(t)$. From eq. (21) we see that the appropriate name for F is velocity drift. Equation (6.21) is not simple, and the reader may consult /33/ for its derivation.

Finally

$$G_3(R',R,t) = \exp\left\{(E-E_L(R))\ t\right\}\quad \delta(R-R') \tag{6.23}$$

Note that G_1 and G_2 are normalized to 1, i.e.,

$$\int G_{1,2}\ (R',R,t)\ dR' = 1 \tag{6.24}$$

whereas G_3 is not normalized.

These particular forms of the individual Green functions has been the reason to choose the particular ordering of operators in eq. (16), because now all integarls appearing in eq. (16.a) may be carried out with the help of the delta functions. Substitute all values (19-23) into eq. (16) to obtain the approximate complete Green function /33/

$$\begin{aligned} G(R',R,t) = (4\pi Dt)^{-3N/2} \exp \left\{ \left[E - (E_L(R)+E_L(R'))/2 \right] t \right\} \\ \exp \left\{ - \left[R' - R - Dt(F(R)+F(R'))/2 \right]^{2} /4Dt \right\} \\ \det \left\{ I - D\, t\, H(R') / 2 \right\} + O(t^3) \end{aligned} \qquad (6.25)$$

where I is the 3Nx3N unit matrix, and H is the 3Nx3N matrix of the spatial derivatives of F

$$H_{pq} = \frac{\partial^2}{\partial R_p \, \partial R_q} \left\{ 2 \ln \;_{T}(R) \right\} .$$

A simpler, less accurate form of the complete Green function is /30/

$$\begin{aligned} G(R',R,t) = (4\pi Dt)^{-3N/2} \exp \left\{ \left[E - (E_L(R)+E_L(R'))/2 \right] t \right\} \\ \exp \left\{ - \left[R' - R - Dt\, F(R) \right]^{2} /4Dt \right\} + O(t^2) \end{aligned} \qquad (6.26)$$

In the next section, when describing the Montecarlo simulation of this problem, we will refer to eq. (26) There is still a simpler form corresponding to the case where importance sampling is not taken into account. This is equivalent to consider the trial wave function to be constant everywhere. Then the drift force is null, and the local energy is just the potential energy, so that

$$\begin{aligned} G(R',R,t) = (4\pi Dt)^{-3N/2} \exp \left\{ \left[E - (V(R)+V(R'))/2 \right] t \right\} \\ \exp \left\{ - \left[R' - R \right]^{2} /4Dt \right\} + O(t^3) \end{aligned} \qquad (6.27)$$

Up to this moment the energy shift appearing in all previous formulae is an arbitrary quantity. Of course, it is in no way related to the expectation value of the hamiltonian in the trial importance sampling function. We already know that when E is equal to the exact ground state energy, then the solution of the differential equations (5) and (7) will stabilize at large times approaching the ground state wave function, namely

$$\Psi(R,t) \longrightarrow C \; \phi_0(R) \text{ , non importance sampling}$$

$$f(R,t) \longrightarrow C \; \Psi_T(R) \; \phi_0(R) \text{ , with importance sampling,}$$

with a coefficient C_0 unknown. This behaviour in the large time limit can be used to compute the energy, using the mixed matrix element

$$E_0 = \frac{\int \Psi_T(R) \; H \; \phi_0(R)}{\int \Psi_T(R) \; \phi_0(R)} \qquad (6.28)$$

In the case of non importance sampling just delete the trial wave function from the above equation. Equation (28) is obvious, because ϕ_0 is the ground state of the hamiltonian. In practical applications it is convenient to act with the hamiltonian operator on the left state. In this form we obtain

$$E_0 = \frac{\int E_L(R) \ \psi_T(R) \ \phi_0(R)}{\int \psi_T(R) \ \phi_0(R)} \cdot \tag{6.29}$$

or, in other words,

$$E_0 = \frac{\int E_L(R) \ \ f(R,t\to\infty)}{\int f \ (R,t\to\infty)} \tag{6.30}$$

This last equation is of deep interest in the DFMC method. After many steps in time we will end up with a set of random points R distributed with the probability distribution function f(R,t). The we may appeal to central limit theorem to get the following estimate for the energy

$$E_0 = \frac{1}{M} \sum_i E_L \ (R_i \ ) \tag{6.31}$$

where $\left\{ R_i, \ \ i=1, \ 2 \ \ldots \ M \right\}$ is the set of random points representing (R,t).

A similar result may be derived in the case of non importance sampling. The only difference is that instead of E_L in eq. (31) there appears V(R).

6.3 Montecarlo solution

After all that mathematical preliminaries we may jump to the practical solution of our problem, namely to eq. (13). The general approach consists in generating a set of random points R'_i, i=1, ... M' , corresponding to the lhs wave function, from the set of random points R_i, i=1, ... M, by means of eq. (13) Note that each point R, i.e. each psip, corresponds to 3N coordinates of the N particles. From each psip R we generate new psips R' by using $G(R',R,\triangle t)$ as a <u>conditional probability</u> distribution. Given that G is not normalized, this means that from and old psip we may end up with none, one or more than one new psips R'. In other words, after each iteration the population of the resulting configuration may grow or may decline. On the average growing or declining is controlled by the value of the energy shift E, this change on the population being the reflect of the change in normalization discussed in eq. (4). After many iterations, the population will be stable only if the energy shift equals the ground state energy. Using eq. (31) one can guess the appropriate value of E.

The Montecarlo algorithm goes through the following steps (this is adapted from Ref. /30/)

0. Preliminaries

Guess a trial function ψ_T, obtain the formula for the local energy, assume a time step and also a value for the energy shift E.

1. Sample the initial distribution, e.g. form $|\psi_T|^2$. This initial distribution is, to a large extent, arbitrary. Usually one starts with a distribution with a very low population, and with an energy shift larger than the ground state energy. In the first iterations the population will grow quite rapidly, and this growing will go towards the correct ground state distribution. Remember that the various components of the starting function do grow more rapidly the smaller their energy is.

Let's call old set this initial distribution, its points being referred as R_i.

2. From each point R_i generate a new point R' using the gaussian part of eq. (26). The new value R' is given by

$$R' = R + D\ t\ F(R) + \chi \tag{6.32}$$

where χ is drawn from the 3N-variate gaussian distribution

$$(4\pi Dt)^{-3N/2} \exp(-\chi^2/4Dt)$$

Note that the gaussian part of eq. (26) is normalized to 1 when integrating with respect to R'. For that reason we obtain at this step one new point from one old point. What we are doing in this step is a displacement from R. This displacement has two parts: a spherically symmetric part χ coming form the Laplace operator term of the hamiltonian, which corresponds to a isotropic diffusion from point R, and a drift DtF(R) directly related to the importance sampling function. The drift force is the log derivative of the trial wave function, and this drift will move the psips towards the region where the importance sampling function is larger.

3. Compute the other term of the Green function, namely the exponential involving the local energies and the energy shift. This part of the Green function is not normalized, and is the responsible of the change in population. This is called the rate term.

4. Branching of psips. The new point has a multiplicity such that on the average equals the value of the rate term. To carry out this step we draw an integer from a distribution which value is the rate term, namely

$$\text{Multiplicity} = \text{Integer}\left\{ \exp\left\{ \left[E-(E_L(R)+E_L(R') \right] t\right\} + \xi \right\} \tag{6.33}$$

where ξ is from U(0,1).

5. If in the previous step the multiplicity is null, then discard the new point R' and go again to step 2 unless the old set is exhausted. Otherwise copy this value R' into the new set as many times as the value of the multiplicity, and add the same number of times the value of $E_L(R')$ to compute the average energy of the iteration, eq. (31). Again go back to step 2.

6. We arrive here when all psips corresponding to the old configuration have evolved. Now we have advanced one time step and have obtained a new configuration. The next task is to advance another time step, so rename the new configuration to the old configuration and go back to step 2. This block should be repeated as many times as needed to reach the asymptotic limit. We may actualize the energy shift E by using eq. (31) to stabilize the population once a critical population has been reached.

When time is long enough we should still advance many time steps to evaluate the expectation value of the energy and control the statistical error. So, the last step is

7. Rename the new set to the old set, test some exiting criterium and go to step 2.

There are some additional comments on this algorithm. Along the iterations, the succesive averages of the energy are strongly correlated if the time step is small, as it should be. To avoid these correlations it is convenient to carry out a block average. The final result will depend on the value of the time step. This is the only systematic error of the algorithm. This error can be corrected by carrying out several runs with different time steps, and extrapolating the result to a null time step value. Finally, this algorithm may be easily modified to deal with the other forms of the time dependent Green function given in eqs. (25) and (27).

6.4 A nucleon in a gaussian potential revisited

As a case example let's consider again the case of a nucleon in a gaussian potential in the same conditions than in section (4.2). Many of the equations necessary have been already deduced in that section, and we will repeat them for convenience:

$$V = - V_0 \exp(-(r/b)^2) \quad , V_0 = -45 \text{ Mev}, b = 2.09 \text{ fm}$$

$$\psi_T = \exp(- \alpha^2 r^2/2) \ , \ \alpha = 0.625 \text{ fm}^{-1} \text{ for variational minimum}$$

$$E_L = V(r) + D\,(3\alpha^2 - \alpha^4 r^2)$$

$$D = \hbar^2/2m = 20.75 \text{ MeV fm}^2$$

and the drift force is given by

$$\vec{F}(r) = -2\,\alpha^2\,\vec{r}$$

and points always towards the origin, where the maximum of the importance sampling function is.

In this case it is quite simple to use the more complex and precise form of the Green function, eq. (25), because of the simplicity of the trial function. The normalized gaussian part of the time dependen Green function is given by

```
C     ============================================================
C     DIFFUSION MONTECARLO WITH IMPORTANCE SAMPLING
C     AND WITH BLOCK AVERAGE
C     ============================================================
      DIMENSION PSIP(4,5000,2)        ! PSIP(1,...), (2,...) (3,...) is point
C                                     ! PSIP(4,...) is Local Energy
      COMMON/RAND/NRAN                ! For Gauss subroutine
      V(X,Y,Z)=-45.*EXP(-(X*X+Y*Y+Z*Z)/4.3681)        ! Potential
C
C     ====      BLOCK 0, Starting parameters ======
C
      TYPE*,' INTEGER FOR RANDOMNESS'
      ACCEPT*,NRAN                    ! Integer for randomness
      TYPE*,' ENTER NUMBER OF BLOCKS, ITERATIONS PER BLOCK AND TIME STEP'
      ACCEPT*,NBLK,NITER,T            ! Blocks, Iterations and Time step
      TYPE*,' ENTER TRIAL ENERGY and TRIAL PARAMETER'
      ACCEPT*,E0,HOALF                ! Importance sampling parameter
      E=E0                            ! Starting energy
      C1=20.75*3*HOALF*HOALF          ! Constants for local energy
      C2=C1*HOALF*HOALF/3.
      D=20.75                         ! Diffusion constant
      STRC=(1.-D*T*HOALF*HOALF)/(1+D*T*HOALF*HOALF)   ! Scaling of Gaussian
      TGAUSS=4*D*T/(1+D*T*HOALF*HOALF)**2             ! G(x,y) parameter
      TYPE*,' ENTER CRITICAL POPULATION AND STARTING POPULATION'
      ACCEPT*,ICRIT,NPOP              ! Critical and starting population
C
C     ==========  BLOCK 1 : Starting configuration ==========
C
      IN=1                            ! IN and IO control the old
      IO=2                            ! and the new PSIP
      EAV=0
      TGAUS=1./HOALF/HOALF            ! Distribution of Trial Fnct Squared
      DO 10 I=1,NPOP                  ! Generate starting PSIP
      CALL GAUSS(X,Y,Z,TGAUS)         ! from variational trial state
      PSIP(1,I,IN)=X
      PSIP(2,I,IN)=Y                  ! Coordinates and ...
      PSIP(3,I,IN)=Z
      PSIP(4,I,IN)=V(X,Y,Z)+C1-C2*(X*X+Y*Y+Z*Z)       ! ... Local energy
10    CONTINUE
C     ==================================================
      EB=0.
      EB2=0.                          ! Block average
      DO 2000 IBLK=0,NBLK
      NUMBER=0                        ! Counts mean population in block
      ENERG=0.                        ! Time iter. average
      DO 100 ITER=1,NITER
      EAV=0                           ! Set average
      JPOP=0                          ! Population of new set
      DO 1000 I=1,NPOP
      X=PSIP(1,I,IN)*STRC             ! ========================
      Y=PSIP(2,I,IN)*STRC             !
      Z=PSIP(3,I,IN)*STRC             !       Step 2
      EL=PSIP(4,I,IN)                 !
      CALL GAUSS(XP,YP,ZP,TGAUSS)     ! Sample new PSIP
      XP=X+XP                         !
      YP=Y+YP                         !
      ZP=Z+ZP                         ! ========================
      ELP=V(XP,YP,ZP)+C1-(XP*XP+YP*YP+ZP*ZP)*C2      ! Local Energy  (3)
      NSONS=INT(EXP(T*(E-0.5*(EL+ELP)))+RAN(NRAN))   ! and Branching (4)
      IF (NSONS.EQ.0) GOTO 1000          ! Descendant rejected
      DO 1010 J=1,NSONS                  ! =========================
      JPOP=JPOP+1                        !
      PSIP(1,JPOP,IO)=XP                 ! Step 5
      PSIP(2,JPOP,IO)=YP                 !
      PSIP(3,JPOP,IO)=ZP                 ! Copy NSONS times sampled point
      PSIP(4,JPOP,IO)=ELP                ! and
      EAV=EAV+ELP                        ! actualize averages
      NUMBER=NUMBER +1                   !
1010  CONTINUE                           ! =========================
1000  CONTINUE
      NPOP=JPOP                          ! =========================
      II=IN                              ! Renaming of OLD and NEW psips
      IN=IO                              ! Step (7)
      IO=II                              !
      E=EAV/JPOP                         ! Update trial energy
      IF(IBLK.EQ.0) GOTO 110             ! Discard averaging in first BLOCK
      ENERG=ENERG+E
110   IF (JPOP.LT.ICRIT) E=E0            ! If population is LOW, increase it
100   CONTINUE
      IF(IBLK.EQ.0) GOTO 2000
      EB=EB+ENERG/NITER
      EB2=EB2+(ENERG/NITER)**2
      PRINT*,IBLK,NUMBER/NITER,ENERG/NITER   ! Print iterations
2000  CONTINUE
      EB=EB/NBLK                         ! Block average
      EB2=EB2/NBLK
      ERR=SQRT((EB2-EB**2)/(NBLK-1))     ! and statistical error
      PRINT*
      PRINT*,' ENERGY =',EB,' ERROR=',ERR
      STOP
      END
c     Insert here subroutine GAUSS, in Fig. (4.1)
```

Figure 6.1. Diffusion Montecarlo with importance sampling.

$$\exp\left[-\left[(1+Dt\alpha^2)\vec{r}\,' - (1-Dt\alpha^2)\vec{r}\right]^2/4Dt\right](1+Dt\alpha^2)^3/(4\pi Dt)^{3/2}$$

This is a normalized gaussian or the form

$$\exp\left[-(\vec{r}\,' - \vec{X})^2 / A\right]$$

where $\vec{X}$ is a scaled coordinate

$$\vec{X} = (1-Dt\,\alpha^2)\,\vec{r}\,/(1+Dt\,\alpha^2)$$

and the gaussian parameter A is

$$A = 4Dt/(1+Dt\,\alpha^2)^2.$$

When $\alpha=0$ all that formulae are the corresponding to the non importance sampling algorithm. Note that the drift is always in the direction of $\vec{r}$, so that step 2 of previous section is simplified. We first scale the actual position r to a new, closer to the origin, position X, and afterwards just add the isotropic gaussian move.

The algorithm described in section (6.3) has been coded in the Fortran program listed if Fig. (6.1). This code follows step by step the mentioned algorithm and with the comments added can be easily followed.

The results corresponding to this simple example are presented in Figure (6.2). We plot the expectation value of the energy computed in the current configuration at every time step. The open circles correspond to the evolution in the absence of importance sampling, and the black circles correspond to the importance sampling case using the optimal harmonic oscillator wave function. The critical population was of 700 points. This means that the energy shift was adjusted to stabilize the population once the number of psips reached this critical value. The time step used was 0.005 MeV^{-1}. The first 60 steps were discarded in obtaining the energy average and the horizontal line shown in the figure is the average of 200 steps more. Its value is -10.9 MeV with a statistical error of 0.1 Mev, to be compared with the exact energy, -10.813 Mev. This is a rather simple and modest calculation, and the results are very gratifying.

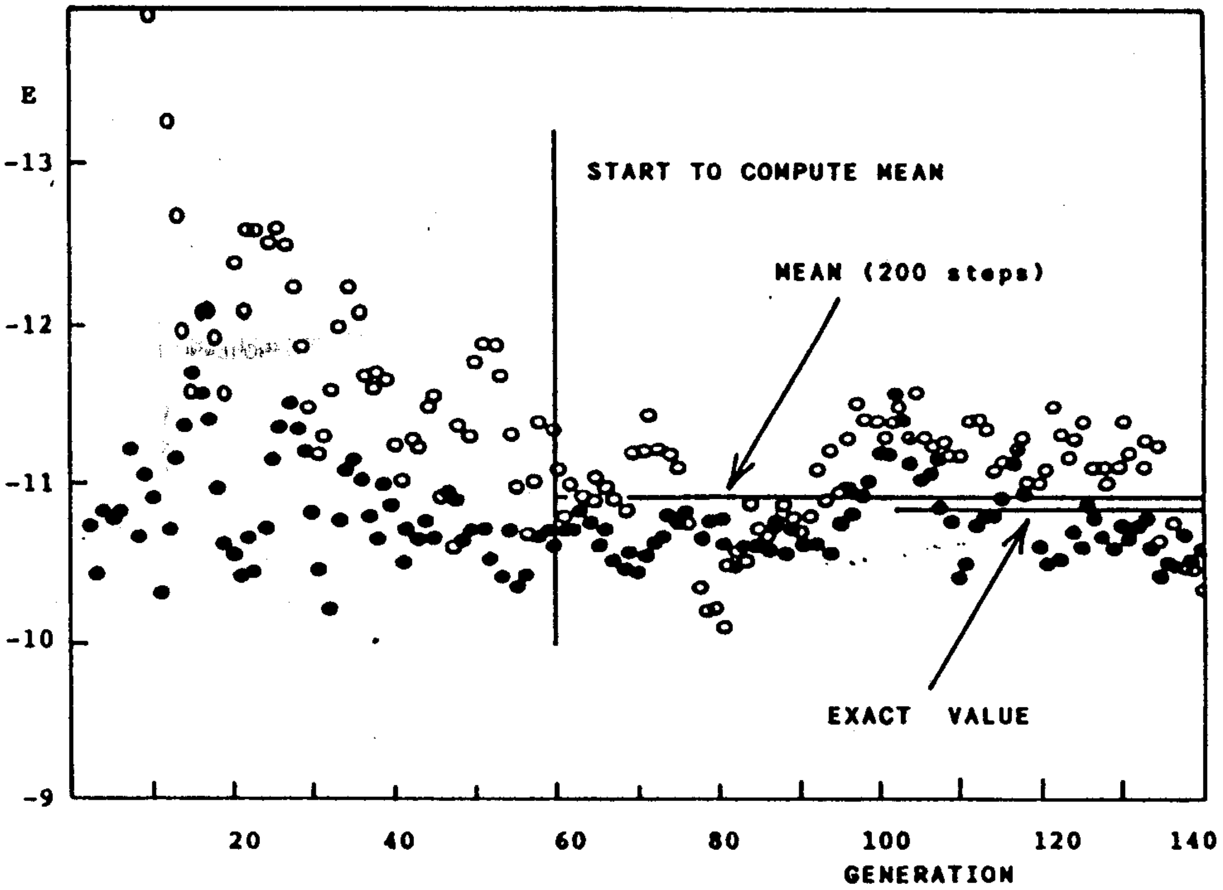


Figure 6.2. Diffusion Montecarlo whithout (open circles) and with (black circles) importance sampling. Energy in MeV.

The calculation with the importance sampling function starts with an already good value for the energy, and the dispersion in the initial iterations is large because the population is small. On the contrary, the values computed without importance sampling show an exponential approach to the correct values in the initial steps. In both cases we observe correlations between the successive steps which suggest the convenience of a block average calculation to get statistical independence. This block averaging is included in the Fortran code of Figure 1.

6.5 Complements

The simple scheme presented in section (6.3) and (6.4) has to be complemented to deal with two common situations, namely the case of extended systems and the case of fermions.

In infinite systems one should use periodic boundary conditions, like in VMC. Whenever a psip exits outside the simulation cell, one should consider its lattice image which entered in the fundamental cell through the opposite wall. Furthermore, the presence of all images must be taken into account when computing the local energy, using an appropriate cutoff or carrying out Ewald's summations.

The problem is not so simple when the system is made up of fermions. Note that the hamiltonian ignores the nature of the particles, and if the previous algorithm is followed as it stands, we will en up with a boson ground state, even if it is not physical. The fermionic character must be imposed externally. The other question is that a fermion wave function is not positive everywhere, and in the regions where it is negative it cannot be interpreted as a probability distribution. This problem was not present in VMC, because the quantity interpreted there as a probability distribution was the wave function squared.

The usual procedure to deal with fermions is the so called fixed node approximation. By means of a trial function Ψ_T one defines a set of nodal surfaces which are the surfaces where Ψ_T is null. These surfaces divide the space in regions where Ψ_T does not change sign. Then one assumes that the exact wave funtion Ψ has the same nodal structure than the trial function, so that $f=\Psi.\Psi_T$ is always positive. These nodal surfaces are assumed to be perfectly absorbing walls: when a psip tries to move to the other side of a nodal surface, it is killed, i.e., is not included in the new generation.

The fixed node algorithm is not exact, but it may be proved /28,30/ that the average of the local energy computed with this approximation is an upper boung to the fermionic ground state energy.

The fixed node approximation can be improved up to some extent, by means of the so called nodal relaxation mechanism. Once fixed node approximation has stabilized, one allows for jumps across the nodal boundaries, but taking track of a signature (+ or -) of the psips. When a psip crosses a nodal surface, its current signature is changed.

After some iterations, every region will have a positive population and a negative population: to compute averages in that region one must add the values corresponding to positive signature psips, and substract the values corresponding to negative signature psips. Because of that, the statistical uncertainty will increase. From the physical point of view what is happening is that a boson population is being superimposed on the fermion population. If many time steps were carried out with node relaxation, we will end up with the boson ground state /27/.

6.6 An application: diatomic molecules

Many of DFMC calculations have been devoted to molecules, the simplest case being diatomic molecules like H_2, LiH, and so on. In all these calculations (as well as in almost all molecular structure calculations) the Born-Oppenheimer approximation is used. This approximation consists in assuming a fixed position for the atomic nuclei, which are static pieces.

When dealing with diatomic molecules one can compute several quantities:

i. The total energy, its meaning being obvious.

ii. The dissociation energy, which is the total energy minus the energy of the non interacting atoms, i.e., when the atomic constituens are at an infinite distance. Chemical measurements determine directly the dissociation energy, corresponding to the equilibrium distance of the nuclei.

iii. Atomic interaction potential, which is the dissociation energy as a function of the distance of the nuclei.

iv. Correlation energy. Is the difference between the total energy and the self-consistent field energy. This quantity has no experimental equivalent, but it is a convenient measure of the goodness of the theoretical approaches.

Calculation in Quantum Chemistry (see Ref. 34 for a full description) have various levels of complexity. The simplest way of carrying out molecular calculations is known as MO-LCAO, and this means that Molecular Orbitals are expressed as a Linear Combination of Atomic Orbitals. The simplest form of these atomic orbitals are known with the name of Slater orbitals. They are hidrogen-like orbitals with an effective charge. The self consistent field (SCF) approach is the expansion of molecural orbitals in a large basis, usually of Slater type. SCF calculations do not include the electron-electron correlation, but they explain the bulk of the dissociation energy. The physically interesting part is the remaining energy, which is called the correlation energy. It may be computed by means of a Configuration Interaction (CI) expansion, or using an explicit dependence on the e-e relative coordinates (Hylleraas form).

In table (6.1) I have collected some recent results corresponding to the LiH molecule, obtained by means of a variety of methods. We cannot enter here in a detailed discussion of these results, but as a general comment we can say that the numbers of Table 1 are somehow relaxing: quite different theories give the correct result within small deviations. Needless to say, any of these calculations are quite costly in terms of computing time.

Table 6.1. Correlation energy of LiH molecule. The SCF energy is 7.987 hartrees (1 hartree = 2 Ry = 27.2 eV). The VMC energy was computed with the importance samplig function of DFMC. Is a poor function, but the nodes are good.

Corr. Energy Milihartrees	Method	Reference
-76.5	CI with Slater orbitals	35
-78.2	CI with elliptic orbitals	36
-80 ± 2	DFMC, fixed node	30
-54 ± 8	VMC	30
-81.5	Coupled Cluster SUB(2)	37
-83.2 0.1	Exact (mixed Theory and Exp.)	38

6.7 Jellium

Jellium is a simplified many body system consisting in N electrons moving in a fixed, uniformly charged background of positive charge which provides electric neutrality. It was proposed 50 years ago by Wigner /39/ as a simplified model of conducting metals. This system may be described in terms of a dimensionless parameter r_s which is the mean interparticle distance measured in units of Bohr radius. High values of r_s correspond to a low density system, and conversely small values fo r_s represent a high density system. A nice feature of Jellium is its expected crystaline configuration (bcc) at low density.

Table 6.2. Correlation energy of jellium in miliRydbergs as a function of r_s, computed by means of various many body theories.

Method	$r_s \rightarrow 0$	1	2	4	10	20	Ref.
DFMC		-120	-90.2	(-63.6)	-37.2	-23.0	27
RPA	62.2 log r_s	-157	-123	-93.6	-61.3	-42.8	41
CC(4)	62.2 log r_s	-122	-90.4	-63.4	-37.0	-23.6	42
FHNC		-138	-98	-67	-37		43
VMC		-122	-87.4	-62.4	-36.3	-22.5	44

This system has been studied by means of almost all many body theories. A set of results of the correlation energy per particle is displayed in Table 6.2. The Random Phase approximation (RPA) and

Coupled Cluster (CC) theories predict the correct behaviour at high density, but as expected RPA fails at low density. A bosonized Fermi Hypernetted Chain (FHNC) theory predicts a high densith behaviour (57 log r_s) which is not bad /40/. However, FHNC is particularly valid at low densities (large r_s) as Table 2 shows.

A nice feature regarding Jellium is its complex phase structure, as predicted by DFMC /27/. At high density the system is paramagnetic (unpolarized fluid), including the region where physical metals should be placed (around r_s=20), and cristalizes at around r_s=100, after passing through a ferromagnetic (polarized) phase. See Ref. /27/ for further details.

7. GREEN FUNCTION MONTECARLO

7.1 Introduction

The basic ideas of GFMC method were developed almost 25 years ago by Malvin Kalos in a nice and simple work /45/, which actually was devoted to the inverse problem: determine the depth of a potential well of prescribed form so as to bind three and four nucleon nuclei at the appropriate energy. The first many body calculation with GFMC for a boson fluid, helium four, appeared several years later, /46/ in 1970, and was limited to a quite small number of particles in the simulation cell. Liquid helium four was subsequently studied with an already complete theroy by Kalos, Levesque and Verlet /47/ coincidindg with the revitalization of many body theories of the last ten years. The role played by VMC and GFMC since 1970 has been decisive to clarify and settle down many of the newly born or newly rediscovered theories, particularly Fermi Hypernetted Chain and Coupled Cluster (exp S) theories.

Further development of GFMC was devoted to many fermion studies /48/ and to few component systems, both fermions /49/ and bosons /50/. At this moment GFMC are a tour de force to large scale (vector) computers. Up to my knowledge, the last GFMC calculation corresponds to liquid helium three /51/.

There exists some nice reviews of GFMC /52-54/, and the reader should consult them to complement or clarify the description of this theory presented here.

GFMC has many features in common with DFMC, and they also share many of the problems and troubles. Actually, GFMC is a time integrated version of DFMC, but DFMC is not a necessary step to describe GFMC. In fact, GFMC is older than DFMC. Again the objective is to solve the many body Schrodinger equation in a stochastic way, by iterating the equivalent integral equation with Montecarlo sampling techniques. The wave function is represented by a set of random points called walkers, and succesive iterations produce generations closer to the ground state wave function.

In this chapter I will skip some of the technical points of GFMC method, and I will pay mainly attention to the conceptual content of this method. As in previous chapters, some examples will be described and documented with the hope of increasing the clarity of the basic ideas.

7.2 Green Functions

Consider the many body problem ($\hbar^2/2m=1$)

$$\left[- \nabla_R^2 + V(R) + V_0\right] \Psi(R) = E \Psi(R) \qquad (7.1)$$

where V_0 is an energy shift so as to have all eigenvalues positive. The value of V_0 is not known a priori, unless the interaction is bounded from below, $V(R) > -C$, with C positive. Then V_0=C can serve as an appropriate shift. In the following V_0 will be included in the potential, so it will disappear from the formulae. The role of the energy shift here is very different of the energy shift in DFMC. Here it is required to have a positive Green function. Below we will introduce a trial energy which will play the same role than the energy shift in DFMC.

The 3N-dimensional Green function satisfies the differential equation

$$(-\nabla_R^2 + V(R)) G(R,R') = \delta(R-R') \qquad (7.2)$$

subjected to the same boundary conditions than eq. (1). So, in finite, localized systems, the wave function drops to zero when any coordinate goes to infinity. In extended systems one should impose periodic boundary conditions, as usual.

From G(R,R') one can determine the wave function by solving the homogeneous integral equation

$$\Psi(R) = E \int G(R,R') \Psi(R') dR' \qquad (7.3)$$

Formally equation (2) is the coordinate representation of the operatorial equation

$$H G = 1 \qquad (7.4)$$

its solution being obviously G=1/H. Note that because of the positivity of H, forced throug the introduction of the energy shift, the solution of eq. (4) is well defined.

Using the complete set of eigenstates of H, $\{\phi_n\}$, eq. (6.3), we may derive the following two forms for the Green function,

$$G = \sum_n |\phi_n\rangle \frac{1}{E_n} \langle\phi_n| \qquad (7.5)$$

and, in coordinate representation

$$G(R,R') = \langle R | G | R' \rangle = \sum_n \phi_n(R) \frac{1}{E_n} \phi_n^*(R') \qquad (7.6)$$

Equation (7.3) may be solved by iteration. Start form an arbitrary state

$$\psi^{(0)} = \sum c_n \phi_n \qquad (7.7)$$

and apply repeteadly the operator $E_T G$ (E_T is a trial energy). Then we obtain after p iterations

$$\psi^{(p)} = \sum_n c_n \left(\frac{E_T}{E_n}\right)^p \phi_n \qquad (7.8)$$

After a large number of iterations the amplitude of ϕ_0 has grown more rapidly than the remaining amplitudes (provided C_0 is not null) and asymptotically only the state ϕ_0, i.e., the exact ground state, will survive. Note that the norm is not maintained along the iterations, and it will grow or decrease when (E_T/E_0) is greater or smaller than 1, respectively. This fact may be used to determine E_0 just by adjusting the trial energy which stabilizes the norm.

The underlying idea of GFMC is the iteration of eq. (3) by means of Montecarlo method. From a generation of points corresponding to iteration number p construct a new generation, corresponding to the nex iteration, by generating new walkers R from the old walkers R' by means of the conditional probability distribution G(R,R').

7.3 Sampling from G(R,R'): Born series

Unfortunately, the Green function is not known in general. So, a pending question of the previous section is how can we sample random points from a distribution function which is not known. The problem is solved by means of a Born series expansion of the complete Green function from an approximate Green function G_0.

Consider the simple problem of a bounded, always attractive potential. We may define a normalized potential v(R) by the equation

$$V(R) = -U\, v(R)$$

with v(R) varying between 0 and 1. Shift both potential and energy by U and use dimensionless coordinates

$$\vec{z} = \alpha \vec{R}, \quad \alpha = (2mU/\hbar^2)^{1/2} \qquad (7.9)$$

Then the Schrodinger equation simplifies to

$$\left[-\nabla^2 + (1 - v(z))\right] \psi(z) = \lambda \psi(z) \qquad (7.10)$$

with

$$\lambda = (U+E)/U \qquad (7.11)$$

Note that $E>0$, and $E<-U$, so that λ is between 0 and 1.

Define now a complete Green function

$$\left[-\nabla^2 + (1-v(Z))\right] G(Z,Z') = \delta(Z-Z') \tag{7.12}$$

and a free Green function

$$\left[-\nabla^2 + 1\right] G_0(Z,Z') = \delta(Z-Z') \tag{7.13}$$

The integral equation to be solved now is

$$\psi(Z) = \int G(Z,Z')\,\psi(Z')\,dZ' \tag{7.14}$$

We will consider eqs. (10), (12) and (13) extended to all space. This condition will be relaxed below.

Some interesting relations may be obtained from these equations. Integrate eq. (10) to obtain the eigenvalue

$$\lambda = \int (1-v(Z)\ \psi(Z)\,dZ \Big/ \int \psi(Z)\,dZ\ . \tag{7.15}$$

Analogously integrate eq. (13) to get the normalization condition of the free Green function

$$\int G_0(Z,Z')\,dZ = 1 \tag{7.16}$$

and also integrate eq. (12)

$$\int (1-v(Z))\ G(Z,Z')\,dZ = 1\ . \tag{7.17}$$

To obtain eqs. (15) to (17) we have integrated by parts the laplacian terms of the required equations. When the boundary is not infinity there will have appeared contributions from the normal derivatives at the boundary surface.

It may be proved that both G and G_0 are positive. Moreover $G(Z,Z')$ is symmetric, and in this particular case G_0 is also symmetric. The symmetry of G_0 will be lost below because of some special boundary conditions.

Next step is to relate G with G_0. Formally

$$G = 1/(T + 1 - v)$$

and

$$G_0 = 1/(T+1),$$

with $T=-\nabla^2$. Use the operatorial identity (this identity is valid when the inverses of A and B are well defined. Again be careful in the case of special boundary conditions)

$$\frac{1}{A} - \frac{1}{B} = \frac{1}{B}(B-A)\frac{1}{A}$$

with $A=T+1-v$ and $B=T+1$, to obtain

$$G = G_0 + G_0\ v\ G \tag{7.18}$$

This is the integral equation relating G and G_0. In coordinate representation eq. (7.18) is

$$G(Z,Z') = G_0(Z,Z') + \int dZ'' \; G_0(Z,Z'') \; v(Z'') \; G(Z'',Z') \qquad (7.19)$$

Solving (18) or (19) iteratively we get the Born series expansion of G,

$$G(Z,Z') = \sum_p G^{(p)}(Z,Z'),$$

with

$$G^{(0)}(Z,Z') = G_0(Z,Z'), \qquad (7.20)$$

and

$$G^{(p)}(Z,Z') = \int dZ_p \; G_0(Z,Z_p) \; v(Z_p) \; G^{(p-1)}(Z_p,Z') \qquad (7.21)$$

The Montecarlo integration algorithm is devised by considering simultaneously the two iterative processes implicit in eq. (14) and in Born series eqs. (20) and (21). The skeleton of the algorithm is the following:

Step 1: Draw an initial state configuration $\{Z_1, Z_2, \ldots Z_N\}$ and assume a value for λ. It is convenient to start with a small population and a value of λ greater than the value corresponding to the ground state, so as to have the population increasing in the correct way.

Step 2: Loop over all walkers of the previous configuration. For each point Z_i of the old set carry out the following steps

a: Draw a new Z from $G_0(Z,Z_i)$. The fisrt time we go through this step it will correspond to the first term of the Born series. Given that G_0 is normalized to 1 we get one new point at this step.

b: Consider now eq. (14). So, with probability λ accept this value Z for the new configuration. Moreover, accumulate $(1-v(Z))$ to have an estimate of the eigenvalue, eq. (15). Even if this Z is not accepted for the wave function of the next generation, go on to the next step. This Z will contribute to the total Green function.

c: Rename Z_i=Z.

d: Compute $v(Z_i)$. We are going to take into account the remaining terms of the Born series, eq. (21). Then sample ξ from U(0,1). If $v(Z_i)$ is less than ξ then go to step e (Born series finished), otherwise go back to step a to consider next term of Born series.

e: Test if i is the last term of the current configuration. If not, repeat from step a for the next walker.

Step 3: At this point one iteration of the integral equation (14) has been completed. We may compute the new estimate for λ and go back to Step 2 a sufficiently large number of times to reach the asymptotic stability and to accumulate statistics.

There still remains to describe how to sample from $G_0(Z,Z')$. The solution of eq. (13) is well known /45/,

$$G_0(Z,Z') = \frac{1}{(2\pi)^{3N/2}} K_{3N/2-1} \left(\left|\vec{Z}-\vec{Z}'\right| \right) / \left|\vec{Z}-\vec{Z}'\right|^{3N/2-1} \qquad (7.22)$$

corresponding to N particles (3N coordinates). K is the imaginary argument Bessel function. Sampling Z when Z' is known corresponds to determine a displacement $\vec{d}$ so that $\vec{Z}=\vec{d}+\vec{Z}'$. The modulus of this displacement follows the above probability distribution, and is isotropically oriented. Kalos /45/ describes the way of carrying out this sampling.

The algorithm presented in this section is quite general, but has some limitations. It can only be applied to bounded attractive potentials, and to a problem extended to all space. It can be directly applied to finite systems with simple interactions, as we will do in the next section.

7.4 A nucleon in a gaussian potential revisited

Consider again the simple example described in sections (4.2) and (6.4), this time with GFMC method as described above.

Carrying out the change of variables (7.9) we have

$$\alpha = 1.4726 \text{ fm} ,$$

$$U = -45 \text{ MeV}$$

$$v(Z) = \exp(-Z^2/9.4730)$$

and the exact eigenvalue is

$$\lambda_{exact} = 0.7597$$

The free Green function may be sampled from the general form eq. (22) for N=1. However, it is convenient here to use the explicit form of the distribution (the Bessel K function 3/2 has a simple form), which in spherical coordinates is

$$G_0 (Z) = Z \exp (-Z) / 4\pi .$$

The method for sampling points from this distribution was described in sect. (2.4), eq. (2.18). This equation gives the modulus of the displacement. The isotropic direction may be sampled in a very simple way. Draw a three-dimensional vector uniformly distributed in a cube of side 2 and centered at the origin. If this vector is outside the sphere of radius 1, then discard it and draw another vector. When the vector is inside the sphere, normalize it and keep it as the desired direction. This method is not very practical in many dimensions, because most of the vectors tried will be outside the sphere. This is due to the fact that the volume of the hypersphere in much smaller than the volume of the hypercube. Kalos /45/ describes a

```
C       GREEN FUNCTION MONTECARLO: a nucleon in Gaussian potential
        DIMENSION R(3,6000,2),X(3),XI(3)
        COMMON /RAND/ NRAN
        DATA ALAM,NUMBER,EIGM,EIGM2,IN,IO/1., 0, 0., 0., 2, 1/
        TYPE*,' INTEGER FOR RANDOM, INITIAL POP., CRITICAL POP AND NITER'
        ACCEPT*,NRAN,NPOP,ICRIT,NITER
        DO 10 I=1,NPOP             ! ======= STEP 1 =======
        DO 10 J=1,3                ! Starting configuration
10      R(J,I,1)=2*RAN(NRAN)-1.    ! generated at random
        DO 100 ITER=1,NITER        ! ===== LOOP OF ITERATIONS =====
        IN=3-IN                    ! Skip from OLD to
        IO=3-IO                    ! NEW configuration
        JPOP=0
        SW=0
        DO 1000 IPOP=1,NPOP        ! ===== LOOP OVER CONFIG =====
        DO 2000 I=1,3
2000    XI(I)=R(I,IPOP,IN)         ! Old walker
1030    CALL GO (XI,X,POT)         ! Sample from G free, == STEP 2a ==
        IF (ALAM.LT.RAN(NRAN)) GOTO 1020! == Step 2b ==, new point rejected
        JPOP=JPOP+1                         ! Accepted, count walkers of new set
        DO 2010 I=1,3
2010    R(I,JPOP,IO)=X(I)                   ! and save for next iteration
        SW=SW+1.-POT                        ! Sampling of 1-v(z)
1020    IF (POT.LT.RAN(NRAN)) GOTO 1000 ! == Step 2d ==
        DO 2020 I=1,3
2020    XI(I)=X(I)                          ! Rename new walker
        GOTO 1030                           ! Go to step == 2a ==
1000    CONTINUE                            ! Born series ended
        EIG=SW/JPOP                         ! Average of eigenvalue over config.
        PRINT*,ITER,JPOP,EIG                ! Monitor the evolution
        NPOP=JPOP
        ALAM=1.                             ! Keep population increasing
        IF(NPOP.LT.ICRIT) GOTO 100
        EIGM=EIGM+EIG              ! Accumulate statistics
        EIGM2=EIGM2+EIG*EIG
        NUMBER=NUMBER+1
        ALAM=EIG                   ! and stabilize population
100     CONTINUE
        EIGM=EIGM/NUMBER
        EIGM2=EIGM2/NUMBER
        PRINT*,' EIGENVALUE=',EIGM,' ERROR=',SQRT((EIGM2-EIGM*EIGM)/(NUMBER-1))
        END
        SUBROUTINE GO (X0, X1,POT)           !Sample form free GREEN function
        DIMENSION X0(3),X1(3)
        COMMON /RAND/NRAN
        R=-ALOG(RAN(NRAN)*RAN(NRAN))        ! Modulus of displacement
100     DO 200 I=1,3                        ! Direction: sample in a cube
200     X1(I)=2.*RAN(NRAN)-1.               ! centered at (0,0,0), with side = 2
        S=X1(1)**2+X1(2)**2+X1(3)**2        ! Compute norm of vector
        IF (S.GT.1.) GOTO 100               ! Outside sphere of radius one. Repeat
        DO 300 I=1,3                        ! Inside, compute coordinates of
300     X1(I)=X0(I)+X1(I)*R/SQRT(S)         ! generated position
        POT=EXP(-(X1(1)**2+X1(2)**2+X1(3)**2)/9.4973)    ! Compute potential
        RETURN
        END
```

Figure 7.1. GFMC algorithm for a nucleon in a gaussian potential.

method appropriate to many dimensions.

The algorithm previously described has been coded in Fortran, and the program is shown in Figure (7.1). It is a literal translation of the explained method and it does not require further comments.

Figure (7.2) plots the results obtained with this program for the caso of a nucleon in the gaussian potential. In general lines the behaviour of the iterations in GFMC is quite similar to the behaviour of the iterations in DFMC. When comparing this figure with Fig. (6.2) note that the scales are different. There is however an important difference, namely the absence of correlations in GFMC between the succesive generations once the system has stabilized. Remember that correlations in DFMC were a direct consequence of the smallness of the time step.

The expectation value of the energy after 200 iterations is λ=0.7594 and the statistical error is 0.001, which corresponds to E=-10.83 ± 0.04. This is a very good result.

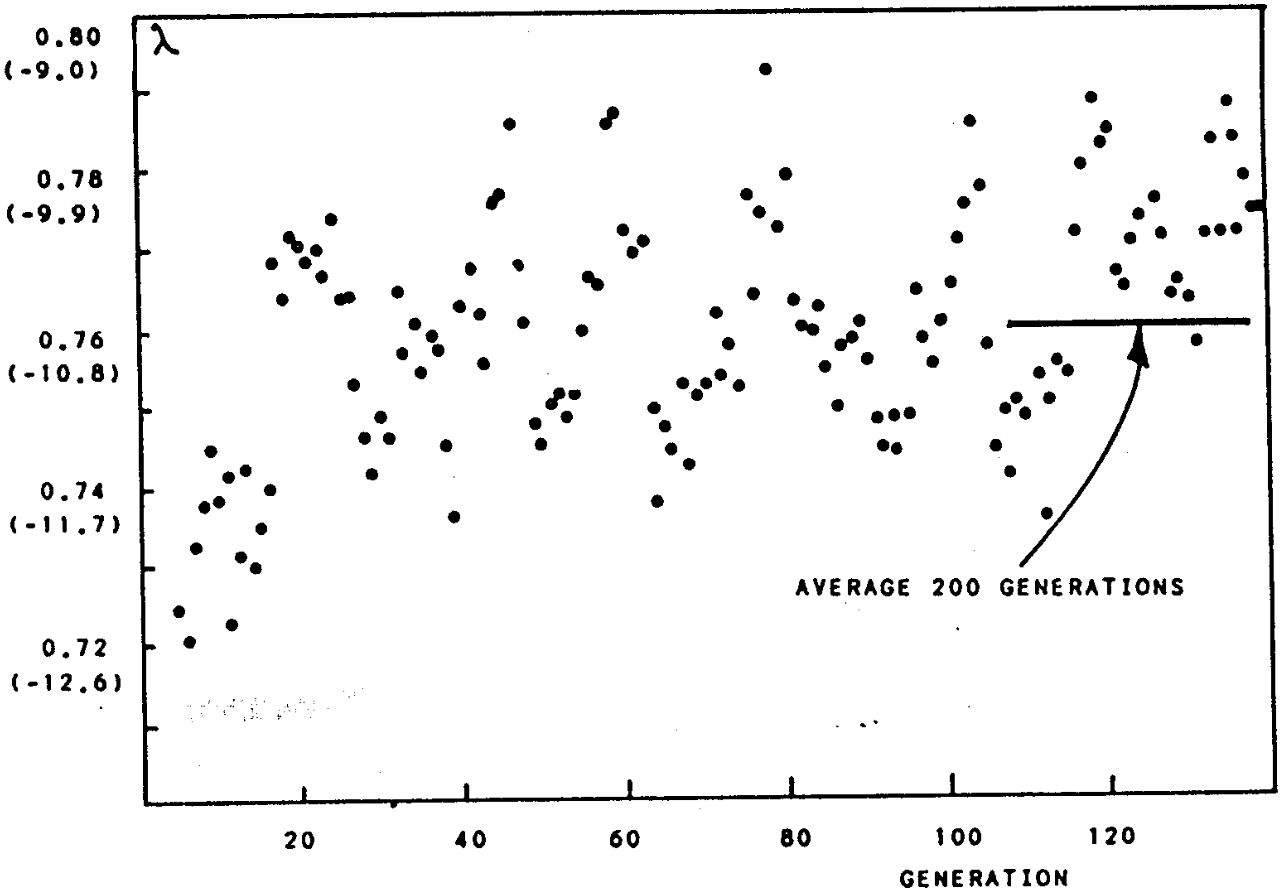


Figure 7.2. The eigenvalue versus the generation number for gaussian potential in GFMC. The values in parenthesis in the vertical scale correspond to the energy in MeV. The horizontal line is the average of 200 generations after the system has stabilized.

7.5 Some complications: unbounded interactions

The previous simple case has shown many of the features of GFMC. However, this method cannot be used with unbounded interactions, so frequent in many body problems.

Kalos, Levesque and Verlet /47/ have developed a method to solve this problem. The basic idea is to establish a Born series expansion for the full Green function in terms of a free Green function defined in a smaller region which does not include the singularities of the potential. Let's describe this method.

Let $G(R,R_0)$ be the full Green function defined in a domain D. Specifically we want to draw random points labelled R from a point labelled R_0. The domain D is the whole physical domain, i.e., all space in finite systems or the fundamental cell with periodic extension in extended systems. G satisfies the equation

$$\left\{ \begin{array}{l} \left[-\nabla_1^2 + V(R_1)\right] G(R,R_1) = \delta(R-R_1) \\ \text{in domain D,} \end{array} \right. \tag{7.23}$$

and is symmetric in R, R_1.

The free or partial Green function is defined in the following form. Let $D(R_0)$ be a subdomain which includes the point R_0, and let U be an upper bound to V(R) in that domain. Note that V(R) is the shifted, always positive potential. The only requirement to define the domain $D(R_0)$ is that it does not include the singularities of the potential. Then define a G_U by the equations

$$\left\{ \begin{array}{l} \left[-\nabla_1^2 + U\right] G_U(R_1,R_0) = \delta(R_1-R_0) \\ G_U(R_1,R_0) \text{ is null if } R_1 \text{ is outside } D(R_0) \end{array} \right. \tag{7.24}$$

Now G_U is no longer symmetric because the dependence of the domain on the point R_0.

One can relate G and G_U by using a familiar technique: multiply on the left eq. (23) times $G_U(R_1,R_0)$, and eq. (24) times $G(R_1,R)$, substract and integrate over R_1 in domain $D(R_0)$. It turns out

$$G(R,R_0) = G_U(R,R_0) + \int_{\text{Surface}} \left[-\nabla_{n_1} G_U(R_1,R_0)\right] G(R,R_1)\, dR_1 + \int_{D(R_0)} \left[U - V(R_1)\right] G_U(R_1,R_0)\, G(R,R_1)\, dR_1 \tag{7.25}$$

In this equation surface means the hypersurface enclosing domain $D(R_0)$, and the gradient operator is the outwards directed normal derivative on this surface.

Equation (25) is again an integral equation similar to eq. (19), even if quite more complicated. It may be solved by a Born series expansion, as before. This Born series is convergent, because the integral of the Kernel is smaller than one. To ascertain this statement integrate eq. (24) in $D(R_0)$ with respect to R_1 to get

$$\int_{Surf.}\left[-\nabla_{n_1} G_U(R_1,R_0)\right] dR_1 + \int_{D(R_0)} U \; G_U(R_1,R_0)\; dR_1 = 1 \qquad (7.26)$$

This is not the integral of the kernel, because it has the quantity U in the second integral instead of $U-V(R_1)$. However $U-V(R_1)$ is smaller than or equal to U in domain $D(R_0)$, so the integral of the kernel is smaller than 1.

Montecarlo integration of eq. (25) is better described if Born series is explicitly written up. Formally eq. (25) is

$$G(R,R_0)=G_U(R,R_0) + \int G(R,R_1)\; K(R_1,R_0)\; dR_1$$

where K contains both surface and volume terms. Iterating we have

$$\begin{aligned} G(R,R_0) = {} & G_U(R,R_0) \\ & + \int G_U(R,R_1)\; K(R_1,R_0)\; dR_1 \\ & + \int G_U(R,R_1)\; K(R_1,R_2)\; K(R_2,R_0)\; dR_2 dR_1 \\ & + \int G_U(R,R_1)\; K(R_1,R_2)\; K(R_2,R_3)\; K(R_3,R_0)\; dR_3 dR_2 dR_1 \; \ldots \end{aligned}$$

where

$$K(R,R') = -\nabla_n\; G_U(R,R') \;\; + U \left\{ 1 - \frac{V(R)}{U} \right\} G_U(R,R')$$

Sampling R given R_0 has two steps. First, sample R from $G_U(R,R_0)$, the first term of Born series. There may result nothing from this sampling, because the volume integral of G_U is smaller than 1, eq. (26). The second step is the sampling from the rest of the series. This may be carried out in a chained way:

1. Put $R_1 = R_0$ (to consider the first term)

2. Put $R_2 = R_1$ (to consider the remaining terms)

3. Define a domain $D(R_2)$, and compute the appropriate constant U for that domain.

4. Sample a new point R_1 form $K(r_1,R_2)$. There are two excluyent cases (see eq. 26):

a. Sample from the surface part of K, with a probability

$$\int \left[-\nabla_n\; G_U(R_1,R_2)\right] dR_1.$$

Call this point r_1 and go back to step 2, or

b. Sample a point in the interior of domain $D(R_2)$ and call it R_1. Now consider the part $(1-V(R_1)/U)$ of the kernel, i.e., continue with a probability $(1-V(R_1)/U)$ going back to step 2. Sampling from Born series will only stop when an interior point is selected and afterwards this point is not accepted.

There are still some formidable pending questions to be answered to make this algorithm practical:

1. How to choose the domain $D(R_1)$

2. How to sample from the volume part of the kernel, G_U, and

3. How to sample form the surface part of the kernel.

I will skip these questions. The reader should consult the works of Kalos, Levesque and Verlet /47/ and Zabolitzky /54/ for a deeper discussion.

7.6 A particle in a box

With the purpose of illustrating some of the ideas introduced in section (7.5) it is convenient to consider the simple case of the problem of a particle in a one dimensional box. The particle is confined to move freely between -1 and +1, and the total Green function is given by the differential equation

$$- \frac{d^2}{dx^2} G(x,x') = \delta(x-x') \tag{7.27}$$

with boundary conditions $G(1,x')=G(-1,x')=0$.

This Green function is well known. It has the form of a triangle with a discontinuity at $x=x'$, i.e.

$$G(x,x') = \begin{cases} (1-x')(1+x)/2 & x < x' \\ (1+x')(1-x)/2 & x > x' \end{cases} \tag{7.28}$$

Define now a G_U Green function and its corresponding domain $D(x')$, see Figure 7.3. There is no potential and the criterium to define $D(x')$ is the size of the box: $D(x')$ extends symmetrically from x' until we arrive to one of the boundaries +1 or -1. G_U will satisfy the same differential equation than G, eq. (27) with the boundary conditions:

$$\left.\begin{array}{l} G_U(1,x') = 0 \\ G_U(2x'-1,x')=0 \end{array}\right\} \quad \text{if } x' > 0 \tag{7.29}$$

or

$$\left.\begin{array}{l} G_U(-1,x') = 0 \\ G_U(1+2x',x')=0 \end{array}\right\} \quad \text{if } x' < 0 \tag{7.30}$$

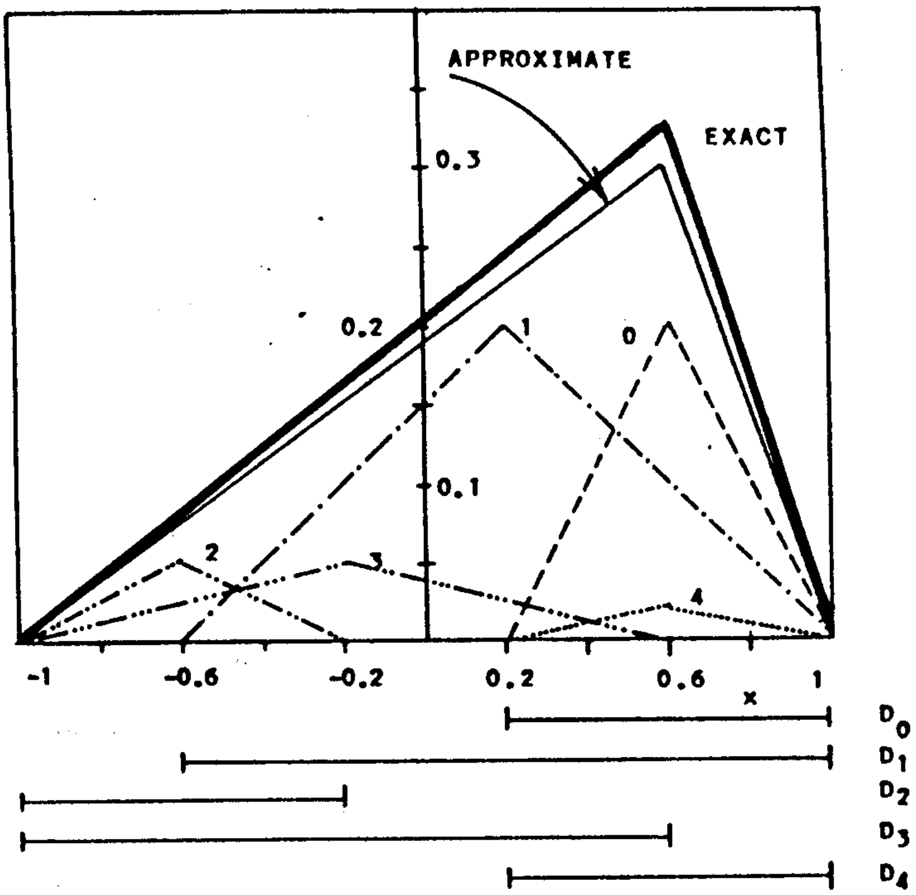


Figure 7.3. Born series for the particle in a box problem. Th segments at the bottom of the figure represent the variou subdomains. See the text for a full description.

The explicit form of G_U is

$$G_U(x,x') \quad \begin{cases} (1+x-2x')/2 & x < x' \\ (1-x)/2 & x > x' \end{cases} \quad \text{if } x' > 0$$

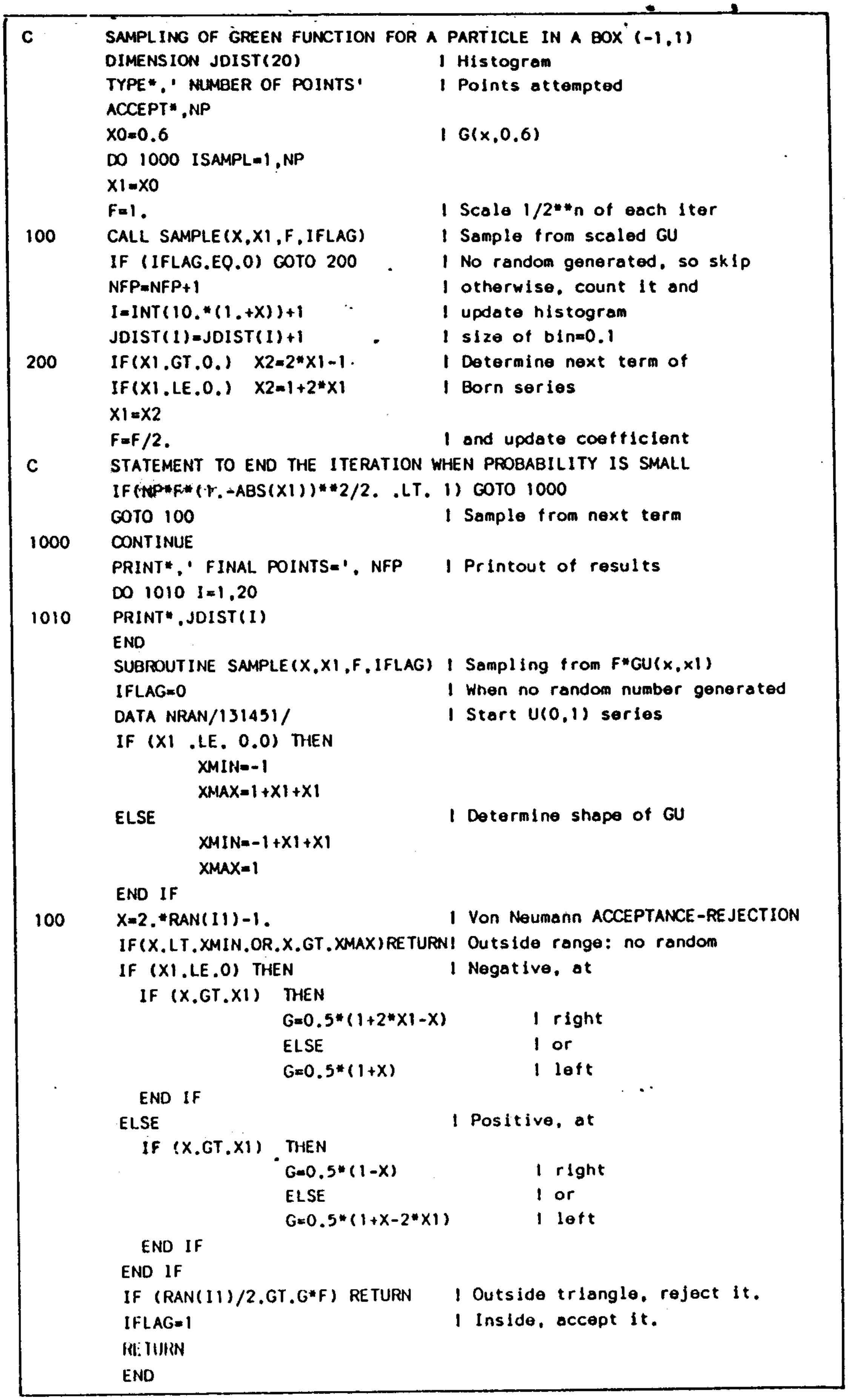
and (7.31

$$G_U(x,x') \quad \begin{cases} (1+x)/2 & x < x' \\ (1+2x'-x)/2 & x > x' \end{cases} \quad \text{if } x' \quad 0$$

Next step is to obtain the integral equation relating G and G Given that there is not a potential, there is not a volume ter Moreover the surface part degenerates to the value at the limit i.e.,

$$G(x_1,x_0) = G_u(x_1,x_0) + \left\{ - \frac{dG_U(x,x_0)}{dx} \; G(x_1,x) \right\}_{x=xsup} \qquad (7.32$$

$$- \left\{ - \frac{dG_U(x,x_0)}{dx} \; G(x_1,x) \right\}_{x=xinf}$$



```
C       SAMPLING OF GREEN FUNCTION FOR A PARTICLE IN A BOX (-1,1)
        DIMENSION JDIST(20)              ! Histogram
        TYPE*,' NUMBER OF POINTS'        ! Points attempted
        ACCEPT*,NP
        X0=0.6                           ! G(x,0.6)
        DO 1000 ISAMPL=1,NP
        X1=X0
        F=1.                             ! Scale 1/2**n of each iter
100     CALL SAMPLE(X,X1,F,IFLAG)        ! Sample from scaled GU
        IF (IFLAG.EQ.0) GOTO 200         ! No random generated, so skip
        NFP=NFP+1                        ! otherwise, count it and
        I=INT(10.*(1.+X))+1              ! update histogram
        JDIST(I)=JDIST(I)+1              ! size of bin=0.1
200     IF(X1.GT.0.)  X2=2*X1-1          ! Determine next term of
        IF(X1.LE.0.)  X2=1+2*X1          ! Born series
        X1=X2
        F=F/2.                           ! and update coefficient
C       STATEMENT TO END THE ITERATION WHEN PROBABILITY IS SMALL
        IF(NP*F*(1.-ABS(X1))**2/2. .LT. 1) GOTO 1000
        GOTO 100                         ! Sample from next term
1000    CONTINUE
        PRINT*,' FINAL POINTS=', NFP     ! Printout of results
        DO 1010 I=1,20
1010    PRINT*,JDIST(I)
        END
        SUBROUTINE SAMPLE(X,X1,F,IFLAG)  ! Sampling from F*GU(x,x1)
        IFLAG=0                          ! When no random number generated
        DATA NRAN/131451/                ! Start U(0,1) series
        IF (X1 .LE. 0.0) THEN
                XMIN=-1
                XMAX=1+X1+X1
        ELSE                             ! Determine shape of GU
                XMIN=-1+X1+X1
                XMAX=1
        END IF
100     X=2.*RAN(I1)-1.                  ! Von Neumann ACCEPTANCE-REJECTION
        IF(X.LT.XMIN.OR.X.GT.XMAX)RETURN! Outside range: no random
        IF (X1.LE.0) THEN                ! Negative, at
          IF (X.GT.X1)  THEN
                        G=0.5*(1+2*X1-X)        ! right
                        ELSE                    ! or
                        G=0.5*(1+X)             ! left
          END IF
        ELSE                             ! Positive, at
          IF (X.GT.X1)  THEN
                        G=0.5*(1-X)             ! right
                        ELSE                    ! or
                        G=0.5*(1+X-2*X1)        ! left
          END IF
        END IF
        IF (RAN(I1)/2.GT.G*F) RETURN     ! Outside triangle, reject it.
        IFLAG=1                          ! Inside, accept it.
        RETURN
        END
```

Figure 7.4. Born series for a particle in a box problem.

when x_1 is negative. Moreover, the exact Green function vanishes at +1 and at -1, so that only one of the two integrated terms will survive. Also, the values of the derivatives of G_U may be computed from its explicit form, eq. (31), and the following 'integral' equation is obtained

$$G(x_1,x_0)=G_U(x_1,x_0) + G(x_1,2x_0-1)/2\ \theta(x_0) + G(x_1,1+2x_0)/2\ \theta(-x_0) \quad (7.33)$$

A Born series may be built up by iterating eq. (33). This is described in Figure (7.3) which corresponds to x_0=0.6. The first domain extends form 0.2 to 1, and the corresponding contribution is the triangle labelled 0. The next iteration is centered at 0.2 and the new domain extends from -0.6 to 1, the corresponding contribution being the triangle labelled 1 (note the factor 1/2 in eq. (33)). Next domain extends from -1 to -0.2,... and so on. This figure includes up to the fifth contribution (which is labelled 4) and the triangle labelled APPROXIMATE is the sum of all that five triangles. The resulting Green function is very close to the exact Green function and, if the series were continued, the exact result will be obtained.

To sample from this series one has to consider all terms of the expansion. Each term is a triangle like eq. (31) times a factor $1/2^p$ corresponding to the order of the iteration. The area of each triangle is $(1-|x|)^2/2$, where x is the center of the subdomain, and for that reason we will only obtain a random number from each individual term with probability $(1-|x|)^2/2^{p+1}$. The area of the total Green function is $(1-x^2)/2$, so that from a run with N trials only $N(1-x^2)/2$ random numbers will result.

Just to complete the present example I include in Figure (7.4) a Fortran program corresponding to this kind of sampling, and the result for x_0=0.6 with 10000 trials is represented in Figure (7.5). A last comment regards to the fact that in this example there is no way to end up, because we always move to a point on the boundary. The Fortran code of figure 4 limits the sampling (statement two lines above statement 1000)

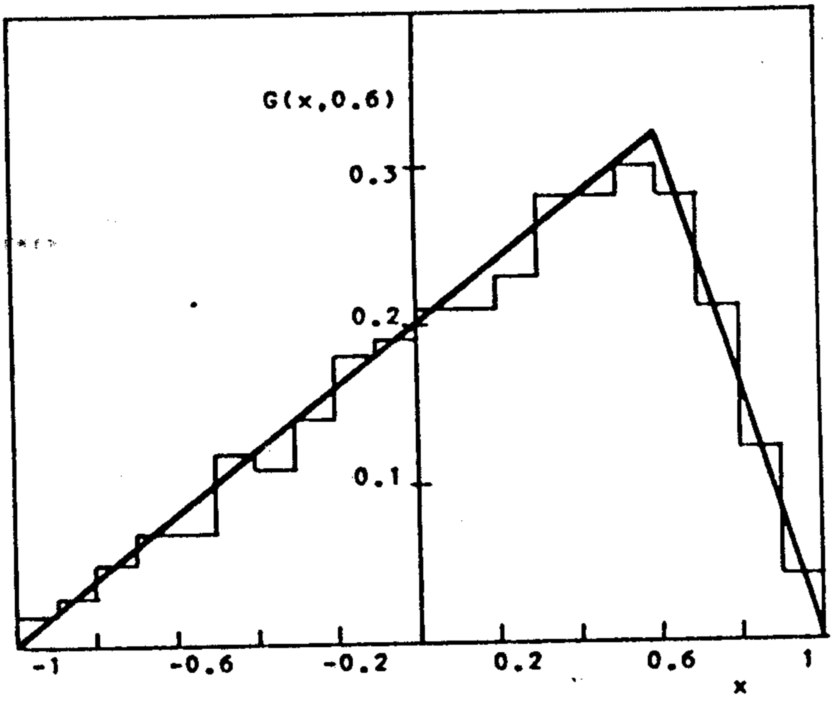


Figure 7.5. Result of Fortran program (7.4) for G(x,0.6)

7.7 Complements: Importance sampling and fermion problems

As in the DFMC case, importance sampling is a very useful tool in GFMC because

i. It is a variance going to zero mechanism, and

ii. It improves the convergence of Born series, by introducing a drift towards the really important regions.

The basic idea is to use the quantity $_T(R)$ (R) as the funtcion to be sampled through the integral equation, like in DFMC case. Actually, importance sampling was firstly developed in GFMC. This new quantity satisfies the integral equation

$$\Psi_T(R)\,\Psi(R) = E \int K(R,R')\;\Psi_T(R')\;\Psi(R')\;dR' \tag{7.34}$$

where K is a modified kernel

$$K(R,R') = \Psi_T(R)\;G(R,R')\;/\;\Psi_T(R') \tag{7.35}$$

One can define a free kernel K_U from the free Green function using the same transformation (35), and obtain an integral equation relating the complete and the free kernels,

$$\begin{aligned}\Psi_T(R)\;G(R,R_0)/\,\Psi_T(R_0) &= \Psi_T(r)\;G_U(R,R_0)/\,\Psi_T(R_0)\\ &+\int_{Surf}\Big[-\,\Psi_T(R_1)\;\nabla_{n_1}G_U(R_1,R_0)/\Psi_T(R_0)\Big]\Big[\Psi_T(R)\;G(R,R_1)/\,\Psi_T(R_1)\Big]\;dR_1\\ &+\int_{Vol}\Big\{U - V(R_1)\Big\}\Big[\Psi_T(R_1)\;G_U(R_1,R_0)/\,\Psi_T(R_0\Big]\Big[\Psi_T(R)\;G(R,R_1)/\Psi_T(R_1\Big]dR_1\end{aligned} \tag{7.36}$$

The stochastic solution of eqs. (35) and (36) follows steps very similar to the steps followed in the non importance sampling method. The more relevant modification is the presence of $\Psi_T(R_1)/\,\Psi_T(R_0)$ each time a G_U function is involved. The role of this correction is easily understood when R_1 and R_0 are rather close points. A Taylor expansion gives

$$\Psi_T(R_1)/\,\Psi_T(R_0) = 1 + (\vec{R}_1-\vec{R}_0)\quad \log\,\vec{\nabla}_T(R_0) \tag{7.37}$$

which shows the drift effect above mentioned.

With this new function one haves the following estimator for the energy

$$E = \frac{\int \Psi(R)\;\;\Psi_T(R)\;E_L(R)\;dR}{\int \Psi(R)\;\;\Psi_T(R)\;dR} \tag{7.38}$$

where E_L is the local energy. As we already know, when the trial function is the exact ground state, then the local energy is a constant and the variance of eq. (38) is zero. In general, we expect to decrease the variance of our estimator by using a reasonable trial function.

As in the case of DFMC, importance sampling helps in dealing with systems of fermions. The troubles related to the antisymmetric nature of the many fermion problem wave function were already briefly described in section (6.5). As in DFMC, the fixed node approximation gives an upper bound to the fermion ground state energy when the mixed estimator eq.(38) is used. The goodness of this upper bound is related to the goodness of the nodal surfaces of the trial function, and not to the function itself.

Fixed node approximation may be followed by nodal relaxation, as in DFMC. This will increase the boson population at each iteration, at the cost of decreasing the fermion population. The consequence is the growing of the variance, so that after some iterations there will remain only fermion noise. Because of that node relaxation can only be carried out a limited number of steps. The energy estimate obtained in this form is called a transient estimate, the reason being obvious.

Two ideas have been put forward to improve the quality of the results for fermions. One of the methods /54,55/ is to keep the total population increasing (using a high value of E in eq. (34)) after relaxing the nodes so as to have a steady fermion population. In practice this means an exponential increase of the population, in other words, an exponential increase of computer time.

Another promising method /53,56,57/ is to include the fermionic character into the Green function itself, instead of being in the wave function. Points are evolved by pairs, one from a positive region and the other from a negative region, and their corresponding Green functions are combined to have a small symmetric component, and an important antisymmetric component. This is a way of compensating the boson background present in the mixed estimator (38). That compensation is exact from the analytic point of view, but only approximate in practice because of the statistical fluctuations. Up to my knowledge, this method has been only used in simple (but not trivial) few body problems /57/.

The description of fermion problems of this section is too short. I suggest the reader to consult Ref. /57/ for a clear and complete analysis of this problem.

7.8 Applications of GFMC: helium systems

Most of GFMC calculations have been devoted to systems of helium atoms. In section (4.4) we have mentioned some results concerning helium-4 with Lennard-Jones potential. The equilibrium energy was found to be a little smaller (absolute energy) than the experimental value, and the discrepancy was associated to the inadequacy of the interaction. Using Aziz /18/ potential, there results a very good value for the binding energy per atom, E(GFMC)=-7.12 0.02 K /17/, to be compared with the experimental value E =-7.14 K.

This result is in a very good agreement with HNC variational calculations /58/ with a wave function including three body correlations, which is -6.96 K. This HNC calculation includes elementary diagrams by means of the so called scaling approximation. A general conclusion of these works is that the description of liquid helium-4 with two- and three- body correlations is appropriate, and that Aziz interaction is a very good interaction to describe liquid helium.

A step forward in the study of helium-4 was the consideration of droplets, i.e., bound clusters of a small number of atoms. A complete an costly GFMC calculation /50/ has been carried out for droplets of varying size, from 3 to 112 atoms per drop. This is an impressive work, both because of the amount of computing time as well as for the beauty of the results. The energy per particle has been fitted to a liquid drop like formula,

$$E(N)/N = E_{vol} + E_{surf}\ N^{-1/3} + E_{curv}\ N^{-2/3}$$

with a volume term (E_{vol}=-7.02 K), a surface term (E_{surf}=18.8 K) and a curvature term (E_{curv}=-11.2 K). The value of the volume term is in a very good agreement with the GFMC calculation mentioned above for the extended system. An important consequence of this work is the confidence it gives to the extrapolation from a finite system to an extended system. Hopefully this extrapolation is also meaningful in nucleon systems.

Another reassuring result is the behaviour of the drop radius with the number of atoms, following the $r=r_0\ N^{1/3}$ low for clusters with 20 or more atoms. Figure (7.6), adapted from Ref. /50/ shows clearly this behaviour.

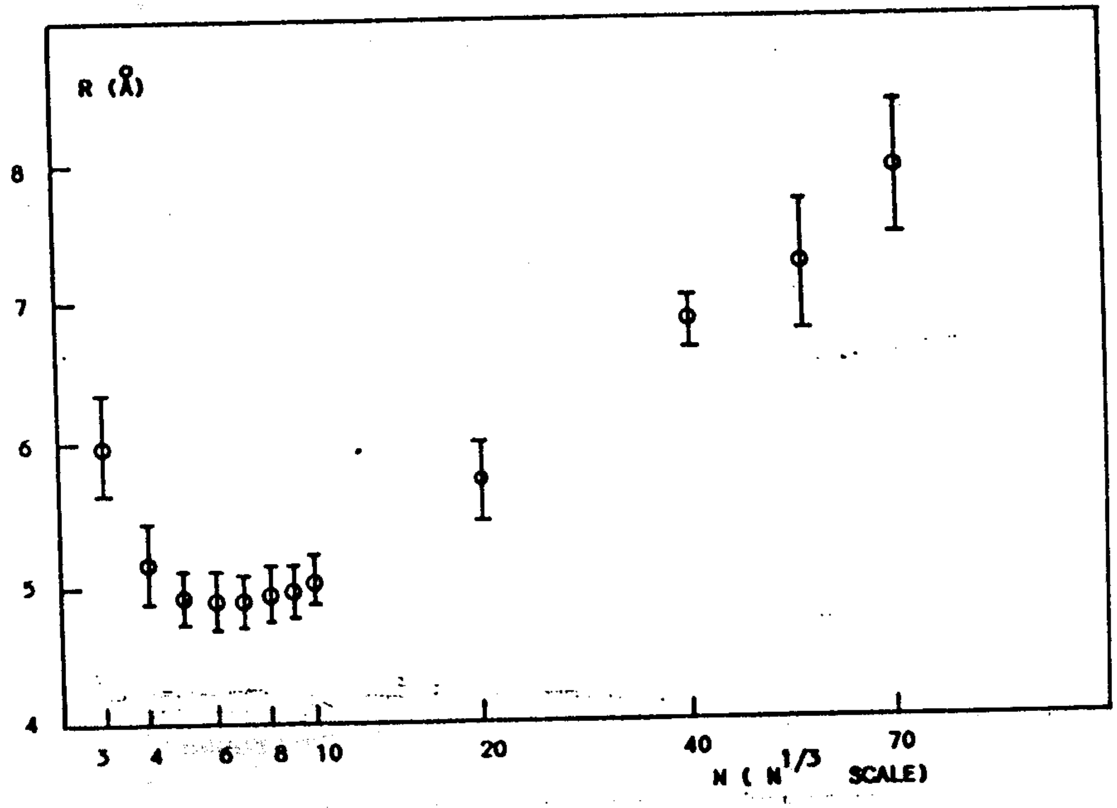


Figure 7.6. Radius of helium-4 droplets versus number of atoms.

Another system which has received very much attention is helium-3, a fermionic system. Variational calculations with Aziz potential have shown the importance of three body correlations, as well as the importance of backflow (formally a momentum dependence correlation). The best variational estimate with VMC method /60/ is E=-1.91 0.03 K, quite low compared with the experimental value of 2.47 0.01 K (I have a hand written number, by Panoff, which is slightly higher, -2.1 K, probably computed with an optimal correlation). This result is also in disagreement with th FHNC+Scaling calculation of Manousakis et al. /61/, which gives -2.36 K. A preliminary, and probably not fully converged, GFMC calculation with fixed node approximation /59/ lowered the VMC result to -2.2 K. This calculation was made with 38 particles only.

The discrepancy between GFMC and FHNC form one side, and with experiment from the other side, has estimulated the improvement of GFMC method. Very recently /51/ some result were presented with a longer GFMC run. The fixed node GFMC obtained was -2.27 0.01 K and -2.32 0.01 K, corresponding to a wave function with the Slater determinant nodes and with the backflow corrected nodes, respectively. A transient estimator lowered the energy to the exciting result of -2.44 0.04 K, but probably this last calculation should be improved.

Attemps have been made to study drops of helium-3 atoms. However, it seems that many atoms are required to form a stable cluster /62/, and there are not still definite results on this system.

8. CLOSING REMARKS

We have arrived to the end of this long trip through Montecarlo forest. Even if the presentation has been somehow superficial in some aspects, I hope that a general view of the power and beauty of Montecarlo method in quantum problems had been reached.

Along the lectures, the emphasis has been mainly directed towards a single quantity, namely the binding energy of the system. Actually with Montecarlo method we obtain much more things than just the energy, for example the wave function. Even if the ground state wave function is obtained in a quite special form, a collection of random points, this information is enough to compute many other properties of the system, particularly the two body distribution function or its Fourier transform, the liquid structure function.

The title of the lectures made a reference to Montecarlo method as a computational technique. Nevertheless, I beleive that one should change this word and write instead theory. In its present form GFMC and DFMC have a solid structure and a form of looking to the problems at hand which convert them in something more than a powerful computational technique.

There are however some disturbing questions behind Montecarlo Many Body theory, and also in other branches of physics. Should one abandone all other theories?. Is Many Body physics reserved to institutions or researchers who have access to really large scale computers?. Even more, some dedicated computers have been constructed (or are being planned) specifically for Montecarlo work, so, are we inventing an Experimental Theoretical Physics?. I beleive I will not be able to answer satisfactorily all these questions. In any case, I feel that feedback among various theories is the most powerful way of progress in physics.

There are still many questions open in Montecarlo study of Many Body systems, and in general in any Many Body theory. Particularly, future work should be focussed in discrete (spin/isospin) degrees of freedom and in the excitation spectrum of extended or finite systems. Probably Montecarlo method has not too much room in these questions, and we have to rely in Correlated Basis Functions or Copuled Cluster theories for the future developments.

ACKNOWLEDGEMENTS

Many people has helped me during the preparation of this work. I would like particularly acknowledge the patience and help of Valdir Aguilera, Ray Bishop, Enrique Buendia, Manuel de Llano, Stefano Fantoni, Bob Panof and Sergio Rosati. The kind hospitality of Leo Violini and the Asociacion pro Centro Internacional de Física made the stay at Bogotá very fruitful and pleasant.

Finantial support from CAICYT (Madrid), under contract 1234/84 is fully acknowledged.

REFERENCES

1. J.M.Hammersley and D.C.Handscomb, "Monte Carlo Methods", Methuen's Monographs (Norwich, U.K., 1967)
2. M.Zelen and N.C.Severo, "Probability Functions" in "Handbook of Mathematical Functions", M.A.Abramowitz and I.A.Stegun Eds. (Dover, New York 1968)
3. G.Marsaglia, "Random Numbers Generation" in "Encyclopaedia of Computer Science", A.Ralston Ed. (Petrocelli/Carter, 1967)

4. K. Binder and D. Stauffer, in "Applications of the Montecarlo Method in Statistical Physics", K.Binder Ed. (Springer, Berlin 1984)

5. M.H.Kalos, Phys. Rev. 128, 1791 (1962)

6. J. von Neumann, Natl. Bur. Stand. Appl. Math. Series 12, 36 (1951)

7. N.Mctropolis, A.W.Rosenbluth, M.N.Rosenbluth,A.H.Teller and E.Teller, Jour. Chem. Phys. 21, 1087 (1953)

8. M. Gardner, "Circo Matemático" (Alianza Ed. Madrid 1983)

9. H. Kahn, Rand Corporation Report R-163 (1949)
R.E. Cutkosky, Jour. Res. Natl. Bur. Stand. 47, 113 (1951)

10. R. Jastrow, Phys. Rev. 98, 1479 (1955)

11. W.W.Wood and F.R.Parker, Jour. Chem. Phys. 27, 720 (1957)

12. A. Munster, "Statistical Thermodynamics" (Academic Press, New York 1969)

13. W.L. McMillan, Phys. Rev. A138, 442 (1965)

14. D. Schiff and L. Verlet, Phys. Rev. 160, 208 (1967)

15. D.M. Ceperley and M.H. Kalos, in "Montecarlo Mehtods in Statistical Physics", K.Binder Ed. (Springer, Berlin 1979)

16. K. Schmidt, M.H. Kalos, M.A. Lee and G.V. Chester, Phys. Rev. Lett. 45, 573 (1980)

17. M.H. Kalos, M.A. Lee, P.A. Whitlock and G.V. Chester, Phys. Rev. B24, 115 (1981)

18. R.A. Aziz, V.P.S. Nain, J.S. Carley, W.L. Taylor and G.T. McConville, Jour. Chem. Phys. 70, 4330 (1979)

19. D. Ceperley, G.V. Chester and M.H. Kalos, Phys. Rev. B16, 3081 (1976)

20. J. Lomnitz-Adler, V.R. Pandharipande and R.A. Smith, Nucl. Phys. A361, 399 (1981)
J. Carlson, R.B. Wiringa and V.R. Pandharipande, Nucl. Phys. A371, 301 (1981)
R.B. Wiringa, J.L. Friar, B.F. Gibson, G.L. Payne and C.R. Chen, Phys. Lett. 143B, 273 (1984)

21. J. Carlson ans M.H. Kalos, Phys. Rev. C32, 2105 (1985)

22. N. Metropolis ans S. Ulam, J. A. Stat. Assn. 44, 335 (1949)

23. M.D. Donsker and M. Kac, J. Res. Natl. Bur. Stand. 44, 551 (1950)

24. J.B. Anderson, J. Chem. Phys. 63, 1499 (1975)

25. J.B. Anderson, J. Chem. Phys. 65, 4121 (1976)

26. J.B. Anderson, J. Chem. Phys. 73, 3879 (1980)

27. D.M. Ceperley and B.J. Alder, Phys. Rev. Lett. 45, 566 (1980)

28. D.M. Ceperley, in "Recent Progress in Many Body Theories", J.G. Zabolitzky, M. de Llano, M.Fortes and J.W. Clark Eds. (Springer, Lect.Not.Phys. 142, Berlin 1981)

29. F. Mentch and J.B. Anderson, J. Chem. Phys. 74, 6307 (1981)

30. P.J. Reynolds, D.M. Ceperley, B.J. Alder and W.A. Lester Jr., J. Chem. Phys. 77,.5593 (1982)

31. J.W. Moskowitz, K.E. Schmidt, M.A. Lee and M.H. Kalos, J. Chem. Phys. 77, 349 (1982)

32. J.B. Anderson, J. Chem. Phys. 82, 2662 (1985)

33. J. Vrbik and S.M. Rohtstein, J. Comput. Phys. 63, 130 (1986)

34. F. L. Pilar, "Elementary Quantum Chemistry" (MacGraw Hill, New York 1968)

35. H. Partridge and S.R. Laughoff, J. Chem. Phys. 74, 2361 (1981)

36. D. M. Bishop and L. M. Cheung, J. Chem. Phys. 78, 1369 (1983)

37. B.Jeziorsky, H.J. Monkhorst, K. Szalewicz and J.G. Zabolitzky, J. Chem. Phys. 81, 368 (1984)

38. K. Szalewicz, B. Jeziorsky, H.J. Monkhorst and J.G. Zabolitzky,

J. Chem. Phys. 83, 5543 (1983)

39. E.P. Wigner, Phys. Rev. 46, 1002 (1934)
40. J.G. Zabolitzky, Phys. Rev. B22, 2353 (1980)
41. R.F. Bishop and K.H. Lurmann, Phys. Rev. B17, 3757 (1978)
42. K. Emrich and J.G. Zabolitzky, Phys. Rev. B30, 2040 (1984)
43. L. Lantto and P.J. Siemens, Nucl. Phys. A317, 55 (1979)
44. D.M. Ceperley, Phys. Rev. B18, 3126 (1978)
45. M.H. Kalos, Phys. Rev. 128, 1711 (1962)
46. M.H. Kalos, Ref. /5/. See also
M.H. Kalos, Jour. Comput. Phys. 2, 257 (1967).
47. M.H. Kalos, D. Levesque and L. Verlet, Phys. Rev. A9, 2178 (1974)
48. D. Ceperley, G.V. Chester and M.H. Kalos, Phys. Rev. B16, 3081 (1977)
49. D.M. Arnow, M.H. Kalos, M.A. Lee and K.E. Schmidt, Jour. Chem. Phys. 77, 5562 (1982)
50. V.R. Pandharipande, J.G. Zabolitzky, S.C. Pieper, R.B. Wiringa and U. Helmbrecht, Phys. Rev. Lett. 50, 1676 (1983)
51. R.B. Panoff, work presented at the 10th International Workshop on Condensed Matter Theories (Argonne, 1986), Proc. to be published by Plenum Press.
52. D. Ceperley and M.H. Kalos, in "Montecarlo Methods in Statistical Physics", K. Binder Ed. (Springer, Berlin 1979)
53. K.E. Schmidt and M.H. Kalos, in "Applications of Montecarlo Method in Statistical Physics", K.Binder Ed. (Springer, Berlin 1984)
54. J.G. Zabolitzky, Prog. Part. and Nucl. Phys. 16, 103 (1986)
55. U. Helmbrecht and J.G. Zabolitzky, Nucl. Phys. A442, 109 (1985)
56. M.H. Kalos, Lect. Notes in Phys. 142, 252 (Springer, Berlin 1981)
57. See ref. 49
58. Q.N. Usmani, S. Fantoni and V.R. Pandharipande, Phys. Rev. B26, 6123 (1982)
59. M.A. Lee, K.E. Schmidt, M.H. Kalos and G.V. Chester, Phys. Rev. Lett. 46, 728 (1981)
60. K.E.Schmidt, M.A. Lee, M.H. Kalos and G.V. Chester, Phys. Rev. Lett. 47, 807 (1981)
61. E. Manousakis, S. Fantoni, V.R. Pandharipande and Q.N. Usmani, Phys. Rev. B28, 3770 (1983)
62. See Ref. /55/