

Chapter 9

Quantum Monte Carlo Methods

Any numerical technique that uses random numbers, tends to be called a '*Monte Carlo*' method, after the famous casino. Here we shall discuss various techniques that have been developed to solve the many-electron Schrödinger equation, which are therefore known collectively as Quantum Monte Carlo methods (QMC). The primary difficulty in solving the many-electron Schrödinger equation arises from the fact that the even the simplest (product) many-electron wavefunction $\Psi(\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_N)$ is a $3N$ -dimensional quantity, and so the calculation of expectation values will require $3N$ -dimensional integrals. We have already seen how the Hartree-Fock method then uses the independent-electron approximation to reduce this to a number of two-electron integrals, i.e. 6-dimensional integrals, each being over 4 basis functions.

But what if we want to go beyond the independent-electron approximation? What if we want to have an exact treatment of dynamical correlation? We will then have to handle these $3N$ -dimensional integrals! What is the best way to do this?

We start by revising standard methods for numerical integration of 1D functions, introduce the technique of Monte Carlo integration, and then show why Monte Carlo integration is superior for higher dimensional functions. We then discuss two particular Monte Carlo techniques that have been developed to solve the many-electron Schrödinger equation with very high levels of accuracy.

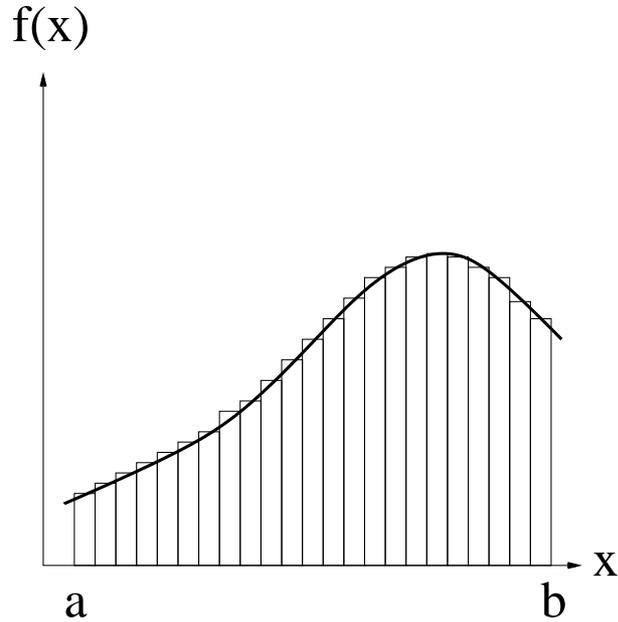


Figure 9.1: Simple integration grid for a smooth function in 1D.

9.1 Revision of numerical integration

9.1.1 Uniform quadrature

There exist many ways to integrate a continuous, bounded 1D function on the interval $[a, b]$ as shown in figure 9.1.

The simplest approach is to define a uniform grid of N points $\{x_i\}$ where

$$x_i = a + ih \quad (9.1)$$

with $h = \frac{(b-a)}{N}$ being the step size, and the index $i : 0 \rightarrow N$. We can then approximate the integral as the sum of the function values on the grid points:

$$\int_a^b f(x) dx \simeq h \sum_i^{N-1} f(x_i) + O(h) \quad (9.2)$$

This is an example of a first-order numerical integration (or *quadrature*) method and has very low efficiency. We can easily generate higher-order algorithms. For example, to generate a second-order method, we replace the constant value $f(x_i)$ that is used across the interval $[x_i, x_{i+1}]$ with a 1st-order polynomial:

$$f(x) \approx f(x_i) + \frac{(x - x_i)}{h} (f(x_{i+1}) - f(x_i)) \quad (9.3)$$

which then gives the *trapezium rule*:

$$\int_a^b f(x) dx \approx h \left(\frac{1}{2}f(x_0) + f(x_1) + f(x_2) + \cdots + f(x_{N-1}) + \frac{1}{2}f(x_N) \right) + O(h^2) \quad (9.4)$$

Similarly, we can extend this to third-order by dividing up each interval into subintervals and doing piecewise quadratic approximations over each subinterval, resulting in *Simpson's rule* (which due to a fortunate cancellation of errors is actually accurate to $O(h^4)$). The list can be extended to arbitrary order.

One useful method is to successively repeat the trapezium rule for intervals of size $h, \frac{h}{2}, \frac{h}{4}$, etc. which results in a series of values of the integral. Fitting this series to a simple polynomial makes it possible to extrapolate the result to zero step size, yielding a very accurate result. This is known as the *Romberg method*.

9.1.2 Gaussian quadrature

An alternative approach is to not choose a uniform grid of points, but rather to choose the points in such a way as to only sample the function at the most important points, and then to add up the function values with appropriate weights at each point. With the above example, we note that a simple change of variable will map the original interval $[a, b]$ onto the interval $[-1, 1]$. We can then exploit the properties of the *Legendre polynomials*, P_l (which arise in the solution of the radial Schrödinger equation for the hydrogen atom), which are orthogonal over this interval:

$$\int_{-1}^1 P_l(x) P_m(x) dx = \delta_{lm} \quad (9.5)$$

Instead of using simple polynomial interpolation of $f(x)$ over the uniform interval $[x_i, x_{i+1}]$ we now use Legendre polynomials with the intervals being non-uniform. An N^{th} order Legendre polynomial has N roots, i.e. N zeroes, and so we choose these as our grid points, with an effective step size $h = \frac{2}{N}$. The resulting *Gauss-Legendre algorithm* is:

$$\int_{-1}^1 f(x) dx \approx \sum_{i=1}^N w_i f(x_i) + O(h^{2N}) \quad (9.6)$$

where w_i are the weights associated with each point and depend on the order of the algorithm.

Note that the accuracy of the method using N points is equivalent to a uniform-grid method using $2N$ points, and the order of the method is also $2N$. Hence this method can achieve very accurate results using fewer function evaluations than uniform grid methods, as long as the function is

reasonably smooth over the interval. However, it is more complex to code and harder to extend to more points.

Note also that other interpolating functions and corresponding weights can be used, which gives rise to other *Gaussian quadrature* schemes.

9.1.3 Higher dimensions

These schemes can be simply extended to higher dimensions. However, there are two major problems:

1. The number of function evaluations required to achieve a given accuracy. If an integral requires N points to get a satisfactory answer in 1D, then it will require N^d points in d -dimensional space. So, 30 function evaluations in 1D become 27000 function evaluations in 3D, etc.
2. The region of integration is defined by a $d-1$ dimensional surface which may be very complex, and can make the creation of an appropriate integration grid difficult.

Both of these difficulties may be overcome by the *Monte Carlo integration* technique.

9.2 Monte Carlo integration

In Monte Carlo integration, we have each point contributing to the integral with a uniform weight, but now we choose the points randomly. Before we discuss why this is a useful way of doing numerical integration, we shall re-cap a little basic probability.

9.2.1 Revision of probability

1. The probability that a *continuous* variable x lies in the range $x \rightarrow x + dx$ is given by $p(x) dx$, where

$$\int_{-\infty}^{\infty} p(x) dx = 1, \quad p(x) \geq 0 \forall x \quad (9.7)$$

which defines the normalisation of $p(x)$.

2. The *mean* value of the variable x is given by

$$\langle x \rangle = \int_{-\infty}^{\infty} p(x) x dx \quad (9.8)$$

and the corresponding *variance* is given by

$$\sigma^2 = \int_{-\infty}^{\infty} p(x) x^2 dx - \langle x \rangle^2 \quad (9.9)$$

3. The average value of any function of x is given by

$$\langle f(x) \rangle = \int_{-\infty}^{\infty} p(x) f(x) dx \quad (9.10)$$

etc.

4. For *discrete* variables, we distinguish between *sample* and *population* statistics. For a sample of size N we find that the sample mean $\langle x \rangle$ is an *unbiased estimator* of the population mean μ :

$$\mu = \langle x \rangle = \frac{1}{N} \sum_{i=1}^N x_i \quad (9.11)$$

but that the sample variance s^2 is a *biased estimator* of the population variance σ^2 and so we have:

$$\sigma^2 = \frac{N}{N-1} s^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \langle x \rangle)^2 = \frac{1}{N-1} (\langle x^2 \rangle - \langle x \rangle^2) \quad (9.12)$$

5. Finally, the *standard error* in the estimate of the mean is given by

$$s_{\langle x \rangle} = \frac{s}{\sqrt{N}} \quad (9.13)$$

and if N is large enough then the value of the mean will be *Normally distributed*, according to the *central limit theorem*.

9.2.2 Evaluating integrals

We now use the *mean value theorem* to evaluate the integral:

$$\int_a^b f(x) dx = (b-a) \langle f \rangle \quad (9.14)$$

That is, we choose the set of N grid points randomly, evaluate $f(x)$ at these random points, and hence make an estimate of the average value of $f(x)$ which therefore leads to a statistical uncertainty in the result

$$\sigma = \frac{(b-a)}{\sqrt{N}} \sqrt{\langle f^2 \rangle - \langle f \rangle^2} \quad (9.15)$$

We therefore have an integration technique that has an error $O(N^{-\frac{1}{2}})$ regardless of the dimensionality of the problem, whereas in d -dimensions the trapezium rule is $O(N^{-\frac{2}{d}})$ and Simpson's rule is $O(N^{-\frac{4}{d}})$. Clearly, the

Monte Carlo method is better than the trapezium rule for $d > 4$ and Simpson's rule for $d > 8$. In many-body quantum mechanics, even the simplest form of the N -body wavefunction is $3N$ -dimensional, and so it can be seen that the Monte Carlo integration technique is invaluable. Even very simple molecules, such as O_2 will have 16 electrons and so be 48-dimensional, and the number of dimensions rises very rapidly with more complex materials.

9.2.3 Boundary conditions

We can also use Monte Carlo integration in situations where the boundary surface is hard to sample randomly. All we need to do is define a sample volume V that includes the region of interest, and extend the definition of $f(x)$ to be zero for points in V that lie outside the region of interest. Including such zero-weighted points will not affect the value of the integral but will increase the estimated error, as the effective number of points is reduced. Obviously, it is best to arrange the sample volume to be as close as possible to the region of interest.

9.2.4 Importance sampling

It is often the case that the contributions to the integral from different sub-volumes within the region of interest vary considerably. If there are small sub-volumes that make a large contribution to the overall integral, then these will only be sampled rarely, and so there will be large statistical errors in the result. It is therefore better to increase the density of sample points in those regions which contribute most to the integral. This is known as *importance sampling*. Obviously, this requires some knowledge of the shape of the integrand before the calculation starts! An alternative technique, known as *adaptive Monte Carlo*, seeks to locate these important sub-volumes "on-the-fly" by probing the function at random points without any prior knowledge of the shape of the function.

9.3 Variational Monte Carlo

9.3.1 Evaluating the energy

The Variational Monte Carlo (VMC) technique combines the variational method discussed in previous lectures with Monte Carlo integration. Our aim is to find the best *many-body wavefunction*, by minimizing the expectation value of the energy:

$$E_{VMC} = \frac{\int \Psi^* \hat{H} \Psi d^{3N}r}{\int |\Psi|^2 d^{3N}r} \quad (9.16)$$

where $\Psi(\mathbf{r}_1, \mathbf{r}_2 \dots \mathbf{r}_N)$ is an N -electron many-body wavefunction. Note that many-body wavefunctions for electrons are necessarily more complex than the simple product of N single-particle wavefunctions due to the requirements of the *Pauli Exclusion Principle*. As discussed before in the Hartree-Fock method, a simple way of making a wavefunction anti-symmetric (i.e. of incorporating the effects of particle *exchange*) is to write it as a *Slater determinant*:

$$\Psi = \frac{1}{\sqrt{N!}} \begin{vmatrix} \phi_1(1) & \phi_1(2) & \cdots & \phi_1(N) \\ \phi_2(1) & \phi_2(2) & \cdots & \phi_2(N) \\ \vdots & & \ddots & \vdots \\ \phi_N(1) & & \cdots & \phi_N(N) \end{vmatrix} \quad (9.17)$$

where $\phi_i(j)$ is a single-particle wavefunction for particle j in state i . As usual, we can expand each single-particle wavefunction in a basis set of known functions. The problem is that as the number of particles increases, it rapidly becomes impossible to perform the integrations exactly using normal quadrature methods. Similarly, if we expand the wavefunction in some basis, it again becomes rapidly impossible to do exact diagonalization of the corresponding matrices.

We therefore seek an alternative approach - that is, we rewrite equation 9.16 in a form that is more amenable for Monte Carlo integration:

$$E_{VMC} = \frac{\int |\Psi|^2 (\Psi^{-1} \hat{H} \Psi) d^{3N}r}{\int |\Psi|^2 d^{3N}r} \quad (9.18)$$

and so if we now define the local energy, E_L as:

$$E_L = \Psi^{-1} \hat{H} \Psi \quad (9.19)$$

then equation 9.18 is now the same form as equation 9.10 with $|\Psi|^2$ being the probability distribution.

We now evaluate this integral M times, with different configurations of all the electrons $\{\mathbf{r}_i\}$ each time, which therefore gives us an approximate value for the energy:

$$E_{VMC} \approx \frac{1}{M} \sum_{j=1}^M E_L(\{\mathbf{r}_i\}) \quad (9.20)$$

where E_L is drawn from the distribution of $|\Psi|^2$.

Note that as we are using Monte Carlo integration, and therefore only finding an approximate value for this integral, it is not unlikely that any given value of E_L will be lower than the true ground state energy. The *variational principle*, that the expectation value of a trial wavefunction is

only an upper bound on the true ground state value, will only apply to the *average* value of E_{VMC} .

We must therefore generate different sets of sample points, i.e. different *configurations*. We do this using the *Metropolis algorithm*:

1. For a given configuration $\{\mathbf{r}_{old}\}$ calculate E_L
2. Generate a new set of points by moving all the points by a random amount
3. Calculate E_L for this new proposed configuration $\{\mathbf{r}_{new}\}$
4. Decide whether to accept or reject this new configuration according to the Metropolis criteria:

- (a) calculate the ratio $p = \left| \frac{\Psi(\{\mathbf{r}_{new}\})}{\Psi(\{\mathbf{r}_{old}\})} \right|^2$
 - i. generate a uniformly distributed random number ξ between $[0, 1]$
 - ii. if $\xi < p$ then accept the new configuration, otherwise reject it.

This algorithm will therefore tend to push the sample points towards the regions of high probability, resulting in importance sampling, with a final distribution of sample points given by $|\Psi_{trial}|^2$. Note that successive values of E_L will continue to vary, resulting in an average value with a corresponding standard error. Obviously, successive configurations will be correlated, and so care must be taken in calculating the variance as successive measurements will not be independent - otherwise the variance will be badly underestimated. See figure 9.2 for an example.

The precision of this answer can therefore be simply improved by increasing the number of configurations sampled.

9.3.2 Improving the wavefunction

To improve the accuracy of our answer we need to be able to improve the functional form of the trial wavefunction. In practice, we usually seek to optimise the wavefunction by minimizing the variance and not the energy, as the variance has a known lower bound of zero unlike the energy for which we do not have a lower bound. There are also efficient and stable algorithms for minimizing objective functions which can be written as a sum of squares.

Note that as $\Psi_{trial} \rightarrow \Psi_{exact}$ the average value, E_{VMC} , will decrease, as will the instantaneous fluctuations, i.e. the variance will tend to zero. This is a result of the *zero variance principle* - since

$$\hat{H}\Psi = E\Psi \tag{9.21}$$

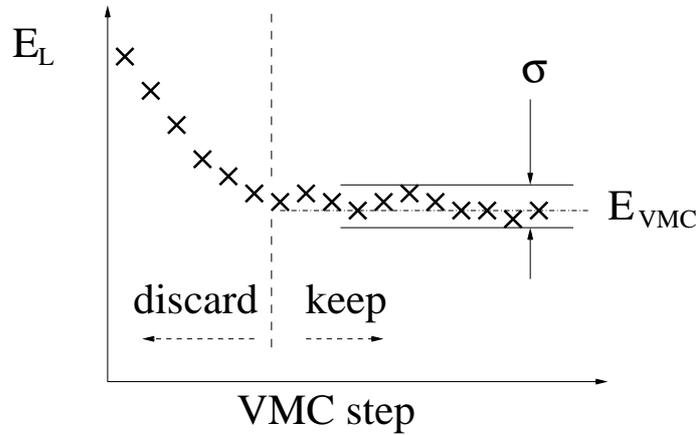


Figure 9.2: Sketch of evolution of E_L during the course of a VMC calculation. The initial phase must be discarded, as this corresponds to configurations that are not very close to $|\Psi_{trial}|^2$. Only the later points are kept for averaging.

then for any eigenstate Ψ of the Hamiltonian we must have

$$\Psi^{-1} \hat{H} \Psi = E \quad (9.22)$$

and so the local energy E_L is the same at all points in configuration space, and therefore E_{VMC} has zero variance if $\Psi_{trial} = \Psi_{exact}$.

We can attempt to include the effects of electron-electron *correlation* by including a *Jastrow function* $J(\alpha)$, writing our correlated many-body wavefunction as a product of a single-particle wavefunction Φ_{trial} (which is not varied) and a Jastrow function which contains a few tens of parameters, $\{\alpha\}$:

$$\Psi(\alpha) = \Phi_{trial} J(\alpha)$$

We can now optimize this wavefunction $\Psi(\alpha)$ by minimizing the variance of the energy:

$$\sigma_E^2(\alpha) = \frac{\int |\Psi(\alpha)|^2 (E_L(\alpha) - E_{VMC}(\alpha))^2 d^{3M}r}{\int |\Psi(\alpha)|^2 d^{3M}r} \quad (9.23)$$

where E_{VMC} is variational and calculated as described above for a fixed $\Psi(\alpha)$:

$$E_{VMC}(\alpha) = \frac{\int |\Psi(\alpha)|^2 E_L(\alpha) d^{3N}r}{\int |\Psi(\alpha)|^2 d^{3N}r} \quad (9.24)$$

We therefore start the VMC calculation with an approximate trial wavefunction from some less accurate *electronic structure method*, such as *Hartree-Fock*, or *Density Functional Theory* and then evaluate E_{VMC} for an initial

set of Jastrow parameters $\{\alpha_0\}$ to derive a value for $\sigma_E^2(\alpha_0)$. We then adjust the parameters $\{\alpha\}$ and re-run the VMC to get a generate a new set of configurations and a new value for $\sigma_E^2(\alpha)$. This is then repeated until we have minimized $\sigma_E^2(\alpha)$ whereupon the wavefunction is as good as it can be given the functional form of the trial wavefunction.

9.4 Diffusion Monte Carlo

The accuracy limitations of VMC can be largely overcome by the Diffusion Monte Carlo (DMC) technique. This has a quite different theoretical foundation, which we shall not discuss in any detail here. In practice, it is often used with VMC as a “first-stage” calculation! We start by considering the time dependent Schrödinger equation in *imaginary time*, $\tau = it$:

$$-\frac{1}{2}\nabla^2\Psi + V\Psi = -\frac{\partial\Psi}{\partial(it)} \quad (9.25)$$

which we can therefore imagine splitting into two parts - a diffusion equation :

$$-\frac{1}{2}\nabla^2\Psi = -\frac{1}{2}\frac{\partial\Psi}{\partial\tau} \quad (9.26)$$

and a rate equation:

$$V\Psi = -\frac{1}{2}\frac{\partial\Psi}{\partial\tau} \quad (9.27)$$

We can then simulate the solution of equation 9.25 using a distribution of *walkers*. Each walker is a point in $3N$ -dimensional configuration space, which moves with a combination of directed drift (equation 9.27) and diffusive motion (equation 9.26). The number of walkers is not constant - they may be killed off in regions of high V or multiplied in regions of low V . The net distribution of walkers will then represent Ψ . This is shown schematically in figure 9.3.

One major problem is that Ψ is not positive-definite, and therefore does not make a very good probability distribution! The rate equation is also badly behaved. The solution to both of these problems is to work with a different probability distribution:

$$f(\mathbf{r}_1, \mathbf{r}_2 \dots \mathbf{r}_N; \tau) = \Psi(\mathbf{r}_1, \mathbf{r}_2 \dots \mathbf{r}_N; \tau) \Phi_{trial}(\mathbf{r}_1, \mathbf{r}_2 \dots \mathbf{r}_N) \quad (9.28)$$

where $\Phi_{trial}(\mathbf{r}_1, \mathbf{r}_2 \dots \mathbf{r}_N)$ is a fixed trial wavefunction, which is typically the final optimized trial wavefunction from a VMC calculation, and Ψ is the “true” unknown many-body wavefunction. By fixing this trial wavefunction, which is already a good approximation to the “true” wavefunction, but allowing it to be multiplied by another function actually has the effect of only

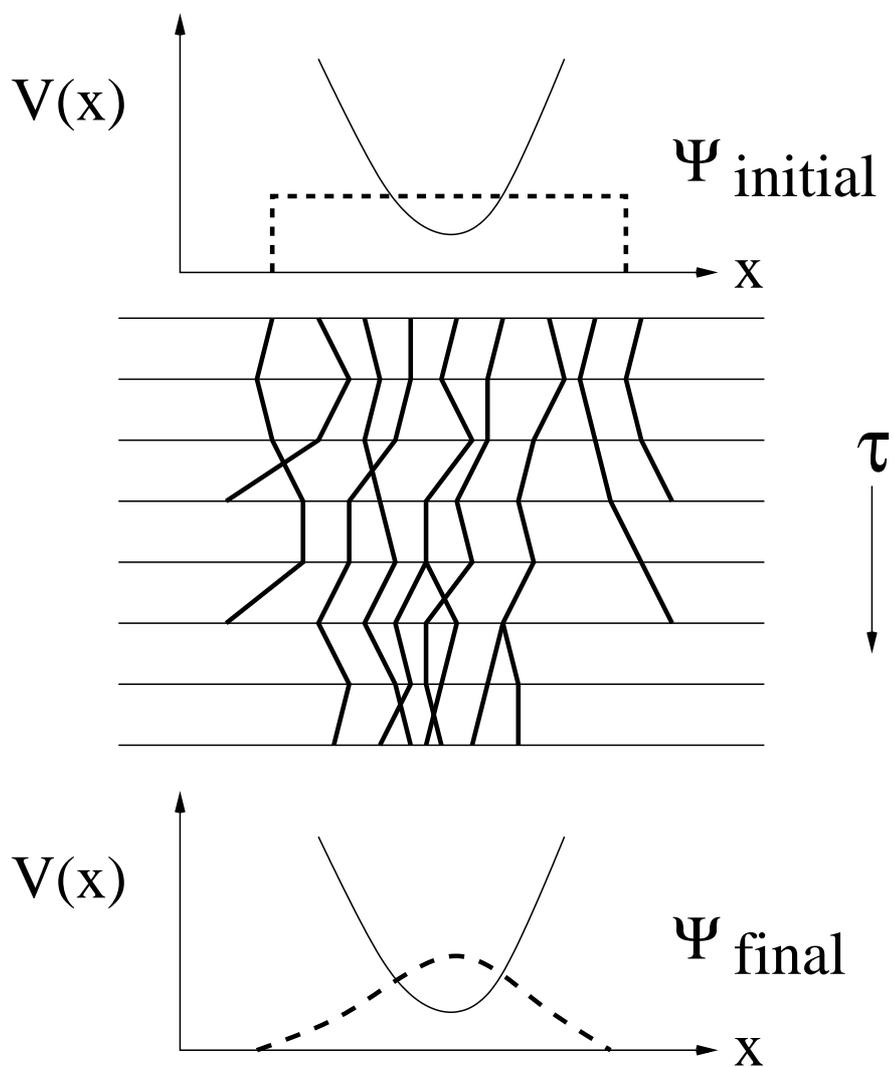


Figure 9.3: Sketch of DMC in action: a simple initial wavefunction Ψ_{init} is represented by an initial density of walkers. The walkers are then moved forwards in imaginary time τ , with some being killed off in regions where V is high and so Ψ ought to be small, and multiplied in regions of low V . The final density of walkers is then a better estimate of the wavefunction, Ψ_{final} .

fixing the points at which the result has a zero value. That is, we have fixed the nodes of the resulting probability distribution, which stops walkers crossing over from regions of positive Ψ to negative Ψ . This is known as the *fixed-node approximation*.

We then calculate the energy as:

$$E_{DMC} = \frac{\int \Psi \Phi_{trial} E_L d^{3N}r}{\int \Psi \Phi_{trial} d^{3N}r} \quad (9.29)$$

$$\approx \frac{1}{M} \sum_{j=1}^M E_L(\{\mathbf{r}_j\}) \quad (9.30)$$

where the local energy, E_L is as before.

9.5 Results

The limiting factor in the accuracy of a VMC calculation is the accuracy of the trial wavefunction. This must be kept fixed as it is used in the Metropolis algorithm to guide the sample points towards the most important regions to yield an accurate integration result. The precision of the VMC calculation can be arbitrarily increased by simply running the calculation for longer, generating more samples and hence reducing the standard error in the results. With DMC, the only significant assumption, which is therefore the limiting factor to the accuracy of a DMC calculation, is the fixed-node approximation.

Note that both VMC and DMC can be applied to excited states as well as the ground state, and gives a good description of a range of properties, not just the energy.

As an example of the abilities of these methods, consider table 9.1. Here we consider the level of treatment of the correlation energy, E_{corr} , which can be considered as the difference between the exact ground state energy (within the Born-Oppenheimer approximation) and the Hartree-Fock energy because, as previously discussed, Hartree-Fock has no treatment of dynamic correlation. We can put some treatment of dynamic correlation into Hartree-Fock by using multiple determinants as in the Configuration Interaction method. Here, we include results from the Coupled Cluster SDT method (CCSDT) (i.e. truncated Configuration Interaction by considering all determinants with single, double and triple excitations but no higher determinants). This is considerably more expensive than single determinant Hartree-Fock and produces a considerable improvement. Note that the correlation energy is not accessible as such in DFT. The results for VMC and DMC calculations show the obvious advantages of these methods. We also include in this table a simple description of the scaling of the various methods with increasing system size (N electrons).

Method	E_{corr}	Scaling
HF	0	$O(N^4)$
DFT	N/A	$O(N^3)$
CCSDT	~75%	$O(N^8)$
VMC	~85%	$O(N^3)$
DMC	~95%	$O(N^3)$

Table 9.1: Ability of different *ab initio* methods to treat dynamical electron-electron correlation, and the scaling of the method with system size.

Method	C	Si	Ge
DFT	7.58	4.84	4.02
VMC	7.36±0.01	4.48±0.01	3.80±0.02
DMC	7.46±0.01	4.63±0.02	3.85±0.02
Expt	7.37	4.64	3.85

Table 9.2: Cohesive energy of different elemental semiconductors in eV/atom.

As another example, in table 9.2 we show the cohesive energies, E_{coh} , of a range of simple semiconductors (the cohesive energy is the binding energy per atom in the bulk material). Whilst the DFT results are already quite accurate (typically within 5% error), the QMC results are even better. Note that this is a calculation that HF has a lot of difficulty with, typically getting up to 50% error!

However, there is a very high computational cost of calculating (a very accurate value for) the energy using QMC techniques. This may be offset by implementing VMC and DMC on a massively parallel computer, which is straightforward to do - more so than HF or DFT! Consequently, QMC techniques have only been taken seriously since the beginning of the 1990's with the advent of sufficiently powerful parallel computers, and even now have not been widely adopted.

One deficiency in any QMC method, that has not been highlighted until now, is that it is not possible to calculate QM forces analytically which makes optimizing the geometry of the atoms very difficult. For this reason, there is still considerable interest in less rigorous, less expensive techniques, such as Hartree-Fock or Density Functional Theory.

9.6 Final comments

A few final points to highlight:

- Monte Carlo integration is much more efficient than either uniform or Gaussian quadrature in higher dimensions.

- VMC is an efficient technique for evaluating the energy of a given many-body wavefunction.
- VMC may also be used to optimize the wavefunction via the Jastrow function.
- Diffusion Monte Carlo is more accurate than VMC but is significantly computationally more expensive.
- Neither DMC nor VMC use the independent-electron approximation and as such are “true” many-body methods, which *may* significantly improve the accuracy of any experimental observables calculated.

9.7 Further reading

- QMC in “Computational Physics” by J.M. Thijssen, chapter 12