Quantum Monte Carlo (QMC)

November 14, 2018
Motivation

We simulate many-body quantum systems by solving the Schrödinger equation stochastically, because there are too many degrees of freedom in anything but trivial systems

Before you begin, you need:

- Hamiltonian (fix particle masses and their interactions)
- Particle statistics: bosons, fermions, or sometimes boltzmannons (forget indistinguishability)
  - A non-relativistic Hamiltonian has no idea if particles are bosons or fermions
- Do we want properties for the ground state, for excited states, or for a finite temperature system?
- Do you really need a first-principles result? DFT is faster than QMC.
QMC methods

• Variational Monte Carlo (VMC)
  – Sample particle positions from a guessed wave function and simulate the consequences in a system with a certain Hamiltonian
  – You may improve your guess (for lower energy or variance), but it’s not automatic

• Diffusion Monte Carlo (DMC)
  – Usually for sampling particle positions in the ground state
  – Numerically improves upon a guessed wave function

• Path Integral Monte Carlo (PIMC)
  – Simulate finite temperature quantum systems
  – Manifest Feynman’s path integral representation of quantum mechanics
  – Also a ground-state variant exists
History

1940’s  E. Fermi: Schrödinger equation can written as a diffusion equation; Physical principles of *Diffusion Monte Carlo (DMC)* discovered.

1942  A. A. Frost: evaluation of the energy of few simple molecules by picking few representative points in the configuration space; *Variational Monte Carlo (VMC)* is born.

1947  N. Metropolis and S. Ulam: Diffusion Monte Carlo developed into an algorithm; “...as suggested by Fermi ...”.

1950’s  R. Feynman: Path integral formulation of quantum mechanics; Physical principles of *Path Integral Monte Carlo (PIMC)* discovered

1962  M. H Kalos: Solved the Schrödinger equation in the form of an integral equation: *Green’s function Monte Carlo (GFMC)* is born.

1964  H. Conroy: Proposed, that points should be picked at random from the configuration space with probability proportional to $\Psi_T^2$, the square of the trial wave function. Application to $H_2^+$, $H^-$, $HeH^{++}$, $He$, $H_2$, and $Li$ - the first application to a fermion system!
1965  W. L. McMillan: Energy of liquid $^4$He; Applied the Metropolis algorithm to sample points with probability proportional to $\Psi_T^2$.
1975  J. B. Anderson: First application of DMC to electronic systems
1980  D. M. Ceperley and B. J. Alder: fixed-node DMC for electron gas
1982  P. J. Reynolds, D. M. Ceperley, B. J. Alder and W. A. Lester, Jr.: fixed-node DMC for molecules
1986  D. M. Ceperley and E. L. Pollock: PIMC calculation of the properties of liquid $^4$He at finite temperatures.
1998  N.V. Prokofev, B. V. Svistunov, and I. S. Tupitsyn: Worm algorithm
2009  G. H. Booth, A. J. W. Thom, and A. Alavi: Fermion MC as a game of life in Slater determinant space
Recent publications: what’s going on

QMC calculations published in October 2018:

- Simulating quantum annealing via projective quantum Monte Carlo algorithms, E. M. Inack
- Estimates of the Quantum Fisher Information in the S=1 Anti-Ferromagnetic Heisenberg Spin Chain with Uniaxial Anisotropy, J. Lambert, E. Sorensen
- Efficient ab initio auxiliary-field quantum Monte Carlo calculations in Gaussian bases via low-rank tensor decomposition, M. Motta et al.
- Quantum simulation for thermodynamics of infinite-size many-body systems by O(10) sites, S. J. Ran et al.
- Many-body effects in (p, pN) reactions within a unified approach, R. Crespo, A. Arriaga, R. B. Wiringa, E. Cravo ...
- Relative energies and electronic structures of CoO polymorphs through diffusion quantum Monte Carlo, K. Saritas, et al.
Computational cost

- Diffusion Monte Carlo (DMC) \( \sim 10 \times \) Variational Monte Carlo (VMC)
- Atomic and molecular physics: fixed-node diffusion Monte Carlo, coupled cluster method CCSD(T)
  - \( N \) electrons, fixed-node DMC as \( N^4 \)
  - CCSD(T) scales as \( N^7 \) \( \Rightarrow \) small systems
    CCSD(T) stands for Coupled Cluster Singles Doubles (Triples), indicating the level of included diagrams; triples are partially included. The method is extremely accurate, but the workload is tremendous.
  - fixed-node DMC scales as \( Z^{5.5-6.5} \) with the atomic number \( Z \): use pseudopotentials?
When is MC effective?

The probability that the point \((x, y)\) is inside the circle is

\[
P(\text{hit circle}) = \frac{\text{Area of a quarter unit circle}}{\text{Area of a box}} = \frac{\pi}{4}. \tag{1}
\]

Very ineffective!

\[
I = \int_0^1 dx f(x). \tag{2}
\]

Estimate

\[
I \approx I_N = \frac{1}{N} \sum_{i=1}^{N} f(x_i), \tag{3}
\]

Multidimensional integrals: The partition function of \(N\) classical atoms is

\[
Z \propto \int dr_1 dr_2 \ldots dr_N \exp \left[ -\beta V(r_1, \ldots, r_N) \right], \tag{4}
\]
When is MC effective?

The variance or the square of the standard deviation of independent samples is

\[
\sigma_I^2 \approx \frac{1}{N} \left[ \frac{1}{N} \sum_{i=1}^{N} f(x_i)^2 - \left( \frac{1}{N} \sum_{i=1}^{N} f(x_i) \right)^2 \right],
\]

(5)

so

\[
\sigma_I \sim \frac{1}{\sqrt{N}}.
\]

(6)

If one has to evaluate the integrand \( N \) times to calculate a \( d \)-dimensional integral, then for each dimension one has \( \sim N^{1/d} \) points, whose distance is \( h \sim N^{-1/d} \). The error for every integration cell of volume \( h^d \) is in the trapezoid rule \( \sim h^{d+2} \), so the total error will be \( N \) times that, \( \sim N^{-2/d} \). For high-dimensional integrals this error diminishes very slowly with increasing number of evaluation points \( N \).
When is MC effective?

Why is MC effective in multidimensional integrations? Heuristically, a 3D cube and a regular grid of $N$ points, take a 2D projection on a square: you see only $N^{2/3}$ points $\Rightarrow$ holes in space, significant parts of space never get any attention.

Ways to do better:

- deterministically: *quasi-Monte Carlo methods*, ”quasi” because the points are not random
- completely random points: Monte Carlo
Correlated samples and MC accuracy

For correlated samples

$$\text{error in mean} = \sqrt{\frac{2\tau_{\text{int}}}{N}} \sigma ,$$  

(7)

where $\sigma$ is the variance of $N$ independent samples. The integrated autocorrelation time $\tau_{\text{int}}$ describes how badly the samples are correlated. The bad news is that close to a critical point $\tau_{\text{int}}$ diverges. This is known as \textit{critical slowing down}, and it tends to make MC calculations slow near phase transitions.

The term ”critical slowing down” can mean two things:

- Decreasing rate of recovery from small perturbations. It signals approaching criticality, such as an ecological disaster. Used as an early-warning mechanism.

- A Monte Carlo method samples the phase space inefficiently due to long-range correlations near criticality. MC simulation spends long time in very similar situations. A more clever algorithm does not suffer from critical slowing down. A typical example is the Ising model simulation, single flip vs. cluster algorithms.
Importance sampling

Probability distribution function (PDF) \( w \),

\[
\int_0^1 dx w(x) = 1 \quad \text{normalization} .
\]  
(8)

write the integral in an equivalent form

\[
I = \int_0^1 dx \, w(x) f(x) \frac{w(x)}{w'(x)}
\]  
(9)

and change variables,

\[
r(x) = \int_0^x dx' w(x').
\]  
(10)

Now we have

\[
\frac{dr}{dx} = w(x),
\]  
(11)

and \( r(x = 0) = 0; \ r(x = 1) = 1 \), so the integral is

\[
I = \int_0^1 dr \frac{f(x(r))}{w(x(r))} .
\]  
(12)
Importance sampling

The ultimate goal of importance sampling is to make the integrand as constant as possible, because with a constant $a$,

$$
\int_0^1 dx \ a \approx \frac{1}{N} \sum_{i=1}^{N} a = a ,
$$

one sample ($N = 1$), say $x = 0.02342$, gives the exact result. You can’t beat this. Integrating

$$
\int_0^1 df(x)
$$

pick $\pi(x)$, which you can sample, to be as close to $f(x)$, so that

$$
\int_0^1 dx f(x) \approx \frac{1}{N} \sum_{i=1}^{N} \frac{f(x_i)}{w(x_i)} , \ x_i \text{ sampled from } \pi(x) ,
$$

and $\frac{f(x)}{w(x)}$ is nearly constant.
Sampling From a Distribution

Define \( r(x) \) as a cumulative function,

\[
r(x) = \int_{-\infty}^{x} dx' w(x') ,
\]

which gives

\[
\frac{dr(x)}{dx} = w(x) .
\]

Apparently,

\[
\int_{0}^{1} dr = \int_{-\infty}^{\infty} dx w(x) .
\]

MC: left integral is evaluated by picking \( N \) random points \( r_i \in U[0, 1] \). The right integral is evaluated by picking points \( x_i \) from distribution \( w \). \( \Rightarrow \) values \( x \) in Eq. (17) have distribution \( w \).
Using inverse functions

A) Discrete distributions
We want to sample integers $k$ with $1 \leq k \leq N$ with probability $P_k$.

1. Construct the cumulative table of probabilities, $F_k = F_{k-1} + P_k$.
2. Sample $r \in U[0,1]$, find $k$ so that $F_{k-1} < r < F_k$. Repeat step 2.

B) Continuous distributions: the mapping method

1. Calculate the cumulative function of $w(x)$,
   \[ F(x) = \int_{-\infty}^{x} dx' \, w(x') \]  
   (21)
2. Sample $r \in U[0,1]$, solve $x$ from $F(x) = r$; the solutions $x$ sample $w(x)$. 

Quantum Monte Carlo (QMC)
Rejection method

Simple case: distribution is limited from above, \( w(x) < M, \ x \in [a, b] \).

1. Draw \( r_1 \in U[a, b] \) and \( r_2 \in U[0, M] \) this picks a random point inside the box around \( w(x) \), see figure

2. If \( w(r_1) \geq r_2 \), accept \( x = r_1 \) as the desired random number, else discard both \( r_1 \) and \( r_2 \). Go to 1.

\[
\begin{align*}
w(x) &= \int_0^M dr_2 \theta(w(x) - r_2) \\
&= \int_a^b dr_1 \int_0^M dr_2 \delta(x - r_1) \theta(w(r_1) - r_2).
\end{align*}
\]
Rejection method

Interpreting the identity as an algorithm:

\[
\theta(w(x) - r_2) = \begin{cases} 
0, & \text{if } r_2 > w(x) \\
1, & \text{if } r_2 < w(x) 
\end{cases},
\]

(23)

with some fixed \( x \), so integrating \( r_2 \) from 0 to \( M \) gives \( w(x) \). Next we use Dirac delta to move \( x \) out of the argument of \( w \), replacing it with \( r_1 \). The pieces can be now interpreted as an algorithm,

\[
w(x) = \int_{a}^{b} dr_1 \int_{0}^{M} dr_2 \delta(x - r_1) \theta(w(r) - r_2).
\]

(24)

The rejection algorithm is just a Monte Carlo integration of the rhs.
Special case: standard normal distribution

**A crude algorithm**

Draw $N$ random numbers $r_i \in U[0, 1]$ and calculate the sum

$$x = \sqrt{\frac{12}{N}} \sum_{i=1}^{N} (r_i - \frac{1}{2})$$  \hspace{1cm} (25)

As $N$ increases this approaches the standard normal distribution, a consequence of the *Central Limit Theorem* (see MC-lecture.pdf for proof):

*If you draw $N$ random numbers from any distribution that itself has a statistical mean $\mu$ and variance $\sigma$, their average is normally distributed with average $\mu$ and variance $\sigma/\sqrt{N}$, as $N \to \infty$.*

Now we draw $N$ random numbers $x_i = r_i - 0.5 \in U[-\frac{1}{2}, \frac{1}{2}]$, a box-shaped distribution $f(x)$ with $\mu = \langle x \rangle = 0$ and

$$\sigma^2 = \langle (x - \langle x \rangle)^2 \rangle = \langle x^2 \rangle = \int_{-\infty}^{\infty} dx x^2 f(x) = \int_{-\frac{1}{2}}^{\frac{1}{2}} dx x^2 = \frac{1}{12},$$

so $\sigma = \frac{1}{\sqrt{12}}$, and hence $\frac{1}{N} \sum_{i=1}^{N} (r_i - \frac{1}{2})$ approaches normal distribution with variance $\frac{1}{\sqrt{12N}}$. Multiply with $\sqrt{12N}$ and you get standard normal distribution ($\sigma = 1$).

This gives a rather good $Normal(x)$ already with small $N$, as $N = 6$ or $N = 12$. The large-$x$ tails are wrong, because you can’t get far with a finite number of steps.
Special case: standard normal distribution

Box-Müller algorithm I

1. Draw $r_1$ and $r_2 \in U[0,1]$
2. Evaluate $\theta = 2\pi r_1$ and $R = \sqrt{-2 \ln r_2}$
3. $(x_1, x_2) = (R \cos(\theta), R \sin(\theta))$
4. Go to step 1.

Proof:

\[
\int_0^1 dr_1 \int_0^1 dr_2 \delta(x_1 - \sqrt{-2 \ln r_1 \cos(2\pi r_2)}) \delta(x_2 - \sqrt{-2 \ln r_1 \sin(2\pi r_2)})
\]

\[
= \int_0^1 dr_1 \int_0^1 dr_2 \delta(r_1 - e^{-(x_1+x_2)^2/2}) \delta(r_2 - \frac{1}{2\pi} \arctan(\frac{x_2}{x_1})) \left| \frac{\partial(r_1, r_2)}{\partial(x_1, x_2)} \right|
\]

\[
= \frac{1}{2\pi} e^{-x_1^2/2} = \frac{1}{\sqrt{2\pi}} e^{-x_1^2/2} \frac{1}{\sqrt{2\pi}} e^{-x_2^2/2} = \text{Normal}(x_1) \text{ Normal}(x_2).
\]

The algorithm simply computes the lhs integrals using MC.

\(^1\)G. Box and M. Müller, Ann. Math. Stat. 38 (1958) 610
Special case: standard normal distribution

Box-Müller algorithm II
Uses the rejection method in order to avoid trig. functions.

1. Draw $r_1$ and $r_2 \in U[0, 1]$
2. Make $R_1 = 2r_1 - 1$ and $R_2 = 2r_2 - 1$, they sample $\in U[-1, 1]$
3. If $R = R_1^2 + R_2^2 > 1$ reject the pair and go to step 1
4. Result:

$$\left( x_1, x_2 \right) = \sqrt{-\frac{2 \ln R}{R}} (R_1, R_2) \quad (27)$$

5. Go to step 1
Fernández–Rivero algorithm

Fernández–Rivero algorithm is a small Maxwell-Boltzmann gas simulation with fictitious energy exchanges (step 5). After thermalization ”particle” velocities \( v \) have normal distribution (step 8)

1. Initialise \( v_i = 1, i = 1, \ldots, N \) (\( N \) is about 1024) and set \( j = 0 \)
2. Draw \( K \) (about 16) values \( \theta_k \in U[0, 2\pi] \).
   Compute \( c_k = \cos(\theta_k) \) and \( s_k = \sin(\theta_k) \).
3. Set \( j = (j + 1) \mod N \), draw \( i \in IU[0, N - 1] \) so that \( i \neq j \)
4. Draw \( k \in IU[0, K - 1] \)
5. Set \( tmp = c_k v_i + s_k v_j \) and \( v_j = -s_k v_i + c_k v_j \)
6. Set \( v_i = tmp \)
7. If still in thermalization go to step 3
8. Result: \( v_i \) and \( v_j \)
9. Go to 3

Here \( IU[0, N] \) means random integer between 0 and \( N \).

Markov Chain Monte Carlo

**Markov chain**

Markov Chain is a chain of states, visited with a rule to pick the next state $x_{i+1}$ from the present one $x_i$. If we go on forever, states are visited with the weight $w(x)$ (normalized PDF is $\pi(x)$).

This Markov chain forms a random walk.

Here $\pi(x)$ is the asymptotic, stationary, or limiting distribution.

**Example 1:** Start the chain from $x_1 = 1$. The rule of the next value is $x_{i+1} = (x_i + 1) \% 5$ (modulo). The chain is $1, 2, 3, 4, 5, 1, 2, 3, 4, 5, 1, ...$. As a result, the weight of each value is $w(x) = 1$, and the PDF is $\pi(x) = 1/5$.

**Example 2:** Start from $x_1 = 0$. The rule is $x_{i+1} = x_i + \sigma \eta$, where $\eta$ is a random number drawn from the standard normal distribution. The weight of each value is $w(x) = e^{-x^2/(2\sigma^2)}$, and the PDF $\pi(x) = 1/(\sigma \sqrt{2\pi})e^{-x^2/(2\sigma^2)}$. 
The MC estimate with $M$ samples of an integral is computed as a sum,

$$\int_{-\infty}^{\infty} dx \frac{1}{\sqrt{2\pi}} e^{-x^2/2} f(x) \approx \frac{1}{M} \sum_{i=1}^{M} f(x_i), \quad x_i \text{ sampled from } \frac{1}{\sqrt{2\pi}} e^{-x^2/2}.$$

(28)

We know how to draw independent samples from a standard normal distribution (Box-Müller, Fernández-Rivero etc.). On the previous slide we stated that the Markov chain with the rule $x_{i+1} = x_i + \sigma \eta$ has

$$\pi(x) = \frac{1}{\sigma \sqrt{2\pi}} e^{-x^2/(2\sigma^2)}.$$
Markov chain vs. direct sampling

Think about computing

\[ \int_{-\infty}^{\infty} w(x) f(x) \approx \frac{1}{M} \sum_{i=1}^{M} f(x_i). \]  

(29)

In principle, you can draw the \( M \) points \( x_i \) from distribution \( \pi(x) \) using

a) Direct sampling of \( \pi(x) \): Always do this if you can.

b) Markov chain \( \{x_1, ..., x_M\} \) that has the limiting distribution \( \pi(x) \).

Here a) is more effective, because the points \( x_i \) are uncorrelated. However, we have algorithms only for sampling a few simple distributions, and many-body wave functions are not among those. Markov chains are inferior in that the points are correlated, but superior in that you can sample an almost arbitrary distribution: All you need is the rule to generate the next point: finding it is our next topic.
Detailed balance

Master equation: The weight (probability if normalized) of a state $S'$ varies as a function of the simulation time (treated as continuous):

$$\frac{d\pi(S')}{dt} = \sum_S P(S \rightarrow S')\pi(S) - \sum_S P(S' \rightarrow S)\pi(S'),$$  \hspace{1cm} (30)

Here $P(S \rightarrow S')$ is the rule to change from $S$ to $S'$, and we need one that gives $\frac{d\pi(S')}{dt} = 0$, because then we get time-independent $\pi(S)$ as the limiting, equilibrium distribution. The detailed balance condition\(^3\) is a solution:

$$P(S \rightarrow S')\pi(S) = P(S' \rightarrow S')\pi(S'),$$  \hspace{1cm} (31)

where $\pi(S)$ is the weight of state $S$.

*The detailed balance condition is a sufficient, but not necessary, condition that the system reaches the correct equilibrium.*

\(^3\)Also known as microscopic reversibility.
Detailed balance

Make the change from state $S$ to $S'$ in two stages:

1. Suggest the change $S \rightarrow S'$ with a transition rule (matrix) $T(S \rightarrow S')$.
2. Accept the change $S \rightarrow S'$ with probability $A(S \rightarrow S')$.

so the rule can be written as

$$P(S \rightarrow S') = T(S \rightarrow S') \cdot A(S \rightarrow S') \quad \text{(32)}$$

"Try"    "Accept"

This way it’s easier to take into account the fact that you may have a nonsymmetric transition rule, $T(S \rightarrow S') \neq T(S' \rightarrow S)$. Requiring that the detailed balance condition is satisfied still leaves an infinite number of possible rules $P(S \rightarrow S')$. Which one is the most effective? Meaning, which rule gives $\pi(S)$ fastest? Let’s assume for a while that $T(S \rightarrow S') = T(S' \rightarrow S)$, so that $P(S \rightarrow S') = A(S \rightarrow S')$, and look at the Metropolis answer in the quest of the most effective rule.
Metropolis algorithm

Original problem: sample states, so that they obey the Boltzmann PDF,

\[ \pi(S) = \frac{1}{Z} e^{-E(S)/T}, \quad k_B \equiv 1. \] (33)

Accept moves from state \( S \) to \( S' \) with probability

\[ A(S \rightarrow S') = \min \left[ 1, e^{-\Delta E / T} \right] = \min \left[ 1, \frac{\pi(S')}{\pi(S)} \right], \] (34)

where \( \Delta E = E(S') - E(S) \). This Metropolis algorithm has two merits:

- The partition function \( Z \) is not involved.
  At equilibrium the master equation gives \( \sum_S P(S \rightarrow S')\pi(S) - \sum_S P(S' \rightarrow S)\pi(S') = 0 \), so any constant factors in \( \pi(S) \) cancels.
- It reaches the equilibrium distribution \( \pi(S) \) fast.
Metropolis-Hastings algorithm

If you try moving right more than left, you should accept left moves more easily. The detailed balance condition with nonsymmetric proposals $T(S \rightarrow S') \neq T(S' \rightarrow S)$:

$$\frac{P(S \rightarrow S')}{P(S' \rightarrow S)} = \frac{T(S \rightarrow S')A(S \rightarrow S')}{T(S' \rightarrow S)A(S' \rightarrow S)} = \frac{\pi(S')}{\pi(S)}. \quad (35)$$

Compensate with acceptance,

$$A(S \rightarrow S') = \min \left[ 1, \frac{T(S' \rightarrow S)}{T(S \rightarrow S')} e^{-\Delta E/T} \right] = \min \left[ 1, \frac{T(S' \rightarrow S)\pi(S')}{T(S \rightarrow S')\pi(S)} \right] \quad (36)$$
Variational Monte Carlo (VMC)

Energy of the trial state $|\varphi_T\rangle$ is

$$E_T = \frac{\langle \varphi_T | H | \varphi_T \rangle}{\langle \varphi_T | \varphi_T \rangle} = \frac{\int dR \varphi_T^* (R) H \varphi_T (R)}{\int dR |\varphi_T (R)|^2}$$

$$= \frac{\int dR |\varphi_T (R)|^2 \left[ \varphi_T (R)^{-1} H \varphi_T (R) \right]}{\int dR |\varphi_T (R)|^2}. \tag{37}$$

The quantity

$$w(R) = \frac{|\varphi_T (R)|^2}{\int dR |\varphi_T (R)|^2} \tag{39}$$

is a good PDF $\Rightarrow$ can be sampled in MC.

Use Metropolis (Markov Chain) to draw samples from $|\varphi_T (R)|^2$

Why: only ratios of weights matter $\Rightarrow \int dR |\varphi_T (R)|^2$ is irrelevant!
Metropolis VMC algorithm

$N$ particles, wf $\varphi_T(R)$, $R = (r_1, \ldots, r_N)$

1. Pick initial configuration $R_1$ and set step size $step$.
3. Draw random displacement vector $d = (d_x, d_y, d_z)$, $d_x$, $d_y$ and $d_z \in U[-\frac{1}{2}, \frac{1}{2}]$.
4. Suggest new position $r'_i = r_i + step \ d$.
5. Compute $ratio = |\varphi_T(R')|^2 / |\varphi_T(R)|^2$.
6. Metropolis question: Accept the move with probability $min[1, ratio]$. If $ratio > 1$ accept move. Else, draw $t \in U[0, 1]$ and if $ratio > t$ accept move. If move was accepted set $r_i = r'_i$, else keep old $r_i$.
7. Pick next particle, set $i = i + 1$. If $i < N$ go to 3.
8. If still in thermalization, go to 3. Adjust $step$ to keep acceptance $\in [40, 60]$.
9. Accumulate averages, print out progress.
Energy optimization

Let’s look at two flavours, energy optimization and variance optimization. **Energy optimization relies on the Rayleight-Ritz variational principle:**

*Among states with the correct symmetry (Bose or Fermi), the ground state has the lowest energy.*

This ensure, that optimization of $\varphi_T(R; \alpha)$ wrt. energy,

$$\frac{\partial E(\alpha)}{\partial \alpha} = 0 \text{ for optimal } \alpha .$$ (40)

Caveats:

- The average $E(\alpha)$ cannot always be determined exactly – in MC it has stochastic error
  $\Rightarrow$ the minimum of the curve $\langle E(\alpha) \rangle$ has stochastic error
  $\Rightarrow$ optimal $\alpha$ has stochastic error
- The many-body wave function is usually not a simple elementary function
  $\Rightarrow$ optimal $\alpha$ gives optimal $\varphi_T(R; \alpha)$ only among a class of functions.
Energy optimization

More serious caveats:

- The many-body wave function cannot be exactly represented by a finite number of samples \( \{\mathbf{R}\} \).

- \( E(\alpha) \) is actually evaluated using a finite sample \( \{\mathbf{R}\} = \{\mathbf{R}_1, \ldots \mathbf{R}_N\} \), so \( E(\alpha) \approx \langle E(\alpha) \rangle_{\{\mathbf{R}\}} \).
  
  - There is no natural lower limit to the energy \( \langle E(\alpha) \rangle_{\{\mathbf{R}\}} \).
    
    \( \Rightarrow \) energy optimization may be unstable, stability may require a very big sample \( \{\mathbf{R}\} \). This usually manifests itself as \( \langle E(\alpha) \rangle_{\{\mathbf{R}\}} \) getting ridiculously low, because the small sample \( \{\mathbf{R}\} \) happened to find a low-energy pithole.

This stability problem is one of the reasons why variance optimization may be more suitable than energy optimization.
Variance optimization

Variance optimization relies on the zero variance principle:

The variance in the variational energy vanishes for a trial wave function, which coincides with any eigenstate of the Hamiltonian.

Proof: Suppose $\varphi_T(R; \alpha) = \phi_i(R)$, the $i$:th eigenstate of $\hat{H}$. The variance of energy is $\sigma^2 = \langle E(\alpha)^2 \rangle - \langle E(\alpha) \rangle^2 = 0$, because

$$\langle E(\alpha)^2 \rangle = \frac{\int dR |\varphi_T(R; \alpha)|^2 E_L(R; \alpha)^2}{\int dR |\varphi_T(R; \alpha)|^2}$$

$$= \frac{\int dR |\phi_i(R)|^2 [\phi_i^{-1}(R) \hat{H} \phi_i(R)][\phi_i^{-1}(R) \hat{H} \phi_i(R)]}{\int dR |\phi_i(R)|^2}$$

$$= \frac{\int dR |\phi_i(R)|^2 E_i^2}{\int dR |\phi_i(R)|^2} = E_i^2$$

(41)

and

$$\langle E(\alpha)^2 \rangle = \left[ \frac{\int dR |\varphi_T(R; \alpha)|^2 E_L(R; \alpha)}{\int dR |\varphi_T(R; \alpha)|^2} \right]^2 = \left[ \frac{\int dR |\phi_i(R)|^2 E_i}{\int dR |\phi_i(R)|^2} \right]^2 = E_i^2.$$  

(42)
Comments on variance optimization:

- Variance – unlike energy – has a natural lower limit, $\sigma^2 \geq 0$.
- Variance-optimized energy $\geq$ energy-optimized energy
Energy vs. variance optimization

Energy optimization and variance optimization are not equivalent. Higher energy result can have lower variance and vice versa:

He atom:
Steps 0-25000: optimized energy, $E = -2.878$, $\sigma^2 = 0.11$.
Steps 25001- : optimized variance, $E = -2.869$, $\sigma^2 = 0.08$. 
Objective functions

Minimize an objective function, one possibility is

\[
\sigma_{\text{opt}}^2(\alpha) = \frac{\sum_{i=1}^{N_{\text{opt}}} w(R_i; \alpha) [E_L(R_i) - E_{\text{guess}}]^2}{\sum_{i=1}^{N_{\text{opt}}} w(R_i; \alpha)},
\] (43)

where the weight function is

\[
w(R_i; \alpha) = \left| \frac{\varphi_T(R_i; \alpha)}{\varphi_T(R_i; \alpha_0)} \right|^2,
\] (44)

and \(R_i\) are sampled using \(\varphi_T(R_i; \alpha_0)\) with a fixed \(\alpha_0\), known as **correlated sampling**, to be explained on the next slide.
Correlated sampling

Direct minimization may be too inaccurate due to statistical fluctuations of data.

The differences of the correlated averages can be much more accurate than the averages themselves.

Energy with parameters $\alpha$:

$$
\langle E(\alpha) \rangle \equiv \frac{\int dR |\varphi_T(R; \alpha)|^2 E_L(R; \alpha)}{\int dR |\varphi_T(R; \alpha)|^2} = \frac{\int dR |\varphi_T(R; \alpha_0)|^2 \left[ \frac{|\varphi_T(R; \alpha)|^2}{|\varphi_T(R; \alpha_0)|^2} E_L(R; \alpha) \right]}{\int dR |\varphi_T(R; \alpha_0)|^2 \left[ \frac{|\varphi_T(R; \alpha)|^2}{|\varphi_T(R; \alpha_0)|^2} \right]} .
$$

- Sample points $R$ using weight $|\varphi_T(R; \alpha_0)|^2$; parameters are $\alpha_0$
- Evaluate (a) $\frac{|\varphi_T(R; \alpha)|^2}{|\varphi_T(R; \alpha_0)|^2} E_L(R; \alpha)$ (b) $\frac{|\varphi_T(R; \alpha)|^2}{|\varphi_T(R; \alpha_0)|^2}$
Correlated sampling

Observations about correlated sampling:

- Variance of individual values $\langle E(\alpha) \rangle$ is larger than sampling from $|\varphi_T(R; \alpha)|^2$ directly.
  $\Rightarrow$ Correlated sampling is no good for absolute energy values.

- Variance of differences $\langle E(\alpha_1) \rangle - \langle E(\alpha_2) \rangle$ can be smaller than what it would be if $\langle E(\alpha_1) \rangle$ was sampled from $|\varphi_T(R; \alpha_1)|^2$ and $\langle E(\alpha_2) \rangle$ was sampled from $|\varphi_T(R; \alpha_2)|^2$:
  Fluctuations "cancel in the same direction", the differences
  \[
  \langle E(\alpha_1) \rangle - \langle E(\alpha_2) \rangle, \tag{46}
  \]
  are more accurate.
  $\Rightarrow$ Correlated sampling is good for hunting energy differences.

This makes correlated sampling suitable for optimization of the trial wave function.
Correlated sampling
Energy optimization with correlated samples

The energy-optimized value is 1.6875, which can be found also analytically – see next page.
Optimization on paper

With a 1S product wf the optimization can be done by hand. e.g. He,

\[
\langle \varphi_T(R) | -\frac{1}{2} \nabla_i^2 | \varphi_T(R) \rangle = \frac{1}{2} \alpha^2
\]  
\[\text{(47)}\]

\[
\langle \varphi_T(R) | -\frac{2}{r_i} | \varphi_T(R) \rangle = -2\alpha
\]  
\[\text{(48)}\]

\[
\langle \varphi_T(R) | -\frac{1}{r_{ij}} | \varphi_T(R) \rangle = \frac{5}{8} \alpha
\]  
\[\text{(49)}\]

so (this is not the local energy, it’s the energy of the trial state)

\[
E(\alpha) = \alpha^2 - 4\alpha + \frac{5}{8} \alpha
\]  
\[\text{(50)}\]

This gives optimized value \( \alpha = 2 - \frac{5}{16} \). Similarly, the first elements give

<table>
<thead>
<tr>
<th>elem</th>
<th>( \alpha )</th>
<th>( E(\alpha) ) (au)</th>
<th>Exact (au)</th>
</tr>
</thead>
<tbody>
<tr>
<td>H</td>
<td>1</td>
<td>-0.5</td>
<td>-0.5</td>
</tr>
<tr>
<td>He</td>
<td>( 2 - \frac{5}{16} \approx 1.6875 )</td>
<td>-2.8476</td>
<td>-2.90372</td>
</tr>
<tr>
<td>Li</td>
<td>( 3 - \frac{5}{8} = 2.375 )</td>
<td>-8.4609</td>
<td>-7.47806</td>
</tr>
</tbody>
</table>
Virial theorem and full optimization

For Coulomb potential, the virial theorem states that

\[ \langle \hat{V} \rangle = -2 \langle \hat{T} \rangle. \] (51)

A good check, but can’t really tell if one trial wave function is better than the other.

**Why not do full optimization?**

Find the normalized wave function of lowest energy:

\[
E[\psi, \psi^*, \lambda] = \int dR \psi^*(R) \hat{H} \psi(R) - \lambda \left( \int dR |\psi(R)|^2 \right)
\]

vary \( \psi^* \):

\[
\delta E = \int dR \delta \psi^*(R) \hat{H} \psi(R) - \lambda \int dR \delta \psi(R)^* \psi(R)
\]

\[
= \int dR \left[ \hat{H} \psi(R) - \lambda \psi(R) \right] \delta \psi(R)^* . \] (52)

Take arbitrary \( \delta \psi(R)^* \), and we are back to Schrödinger equation

\[
\hat{H} \psi(R) = \lambda \psi(R) \quad \text{Harharhar!} \] (53)
Cusp conditions

In the local energy, infinities of the potential energy should be cancelled by those of the kinetic energy\(^4\). These happen if

- two electrons are too close
- electron get too close to nucleus

Concentrate on two electrons, keeping the rest fixed: relative \( \mathbf{r} = \mathbf{r}_1 - \mathbf{r}_2 \) and center-of-mass \( \mathbf{R}_{CM} = \frac{1}{2}(\mathbf{r}_1 + \mathbf{r}_2) \). Since (see lecture notes for details)

\[
\nabla_1^2 + \nabla_2^2 = 2\nabla_{\mathbf{r}}^2 + \frac{1}{2}\nabla_{\mathbf{R}_{CM}}^2 ,
\]

the Hamiltonian is

\[
\hat{\mathcal{H}} = - \left( \frac{1}{4} \nabla_{\mathbf{R}_{CM}}^2 + \nabla_{\mathbf{r}}^2 \right) - \frac{1}{2} \sum_{i=3}^{N} \nabla_i^2 + V(\mathbf{r}, \mathbf{R}_{CM}, \mathbf{r}_3, \ldots, \mathbf{r}_N) .
\]
Cusp conditions

Try a trial wf of Jastrow type,

\[ \Psi(R) = \Psi(r, R_{cm}, r_3, \ldots, r_N) = e^{-u(r)} f(r) Q(R_{cm}, r_3, \ldots, r_N), \quad (56) \]

where

1. \( f(0) \neq 0 \), if the spins at \( r_1 \) and at \( r_2 \) are anti-parallel
2. \( f(0) = 0 \), and \( f(-r) = -f(r) \), if the spins at \( r_1 \) and at \( r_2 \) are parallel

The potential part diverges as 1/r, so we want to keep this \( h(r) \) finite:

\[ h(r) = \frac{1}{e^{-u(r)} f(r)} \left( -\nabla^2 r + \frac{1}{r} \right) e^{-u(r)} f(r). \quad (57) \]

Expand (leave out the arguments \( r \) ),

\[ h = \frac{2u'}{r} + u'' - u'^2 + \frac{1}{r} + 2u' \frac{1}{f} \cdot \nabla r f - \frac{1}{f} \nabla^2 f, \quad (58) \]

where \( u' \equiv du/dr \). Look at the two cases separately.
Cusp conditions

Case 1: anti-parallel spins, $f(0) \neq 0$

Only 1. and 4. terms are singular,

\[
h = \frac{2u'}{r} + u'' - u'^2 + \frac{1}{r} + 2u' \frac{1}{f} \frac{r}{r} \cdot \nabla_r f - \frac{1}{f} \nabla^2_r f ,
\]

so we set

\[
\frac{2u' + 1}{r} \text{ finite },
\]

so the cusp condition is

\[
\left. \frac{du}{dr} \right|_{r=0} = -\frac{1}{2} .
\]
Cusp conditions

Case 2: parallel spins, \( f(0) = 0 \), and \( f(-r) = -f(r) \)

Taylor expansion of \( f(r) \) around origin is

\[
f(r) = a \cdot r + O(r^3),
\]

which has only odd terms because \( f(-r) = -f(r) \). Close to origin we get

\[
h \approx \frac{2u'}{r} + u'' - u'^2 + \frac{1}{r} + 2u' \frac{1}{a \cdot r} \frac{r}{r} \cdot a = \frac{4u'}{r} + \frac{1}{r},
\]

so the cusp condition is

\[
\left. \frac{du}{dr} \right|_{r=0} = -\frac{1}{4}.
\]
Diffusion Monte Carlo (DMC)

Schrödinger equation in imaginary time ⇒ diffusion equation

\[- \frac{\partial |\psi(\tau)\rangle}{\partial \tau} = \hat{\mathcal{H}} |\psi(\tau)\rangle .\] (65)

formal solution is (if \(\hat{\mathcal{H}}\) does not depend on \(\tau\))

\[|\psi(\tau)\rangle = e^{-\hat{\mathcal{H}} \tau} |\psi(0)\rangle \quad \text{projection} .\] (66)

The eigenstates \(|\phi_i\rangle\) of \(\hat{\mathcal{H}}\) are solutions of

\[\hat{\mathcal{H}} |\phi_i\rangle = E_i |\phi_i\rangle, \quad E_0 < E_1 < E_2 < \ldots .\] (67)

We can expand any \(N\)-body state in the basis of eigenstates of \(\hat{\mathcal{H}}\),

\[|\psi(0)\rangle = \sum_i c_i |\phi_i\rangle ,\] (68)

so

\[|\psi(\tau)\rangle = \sum_i c_i e^{-E_i \tau} |\phi_i\rangle .\] (69)
Diffusion Monte Carlo (DMC)

Stabilise the evolution by subtracting a trial or reference energy $E_T \approx E_0$:

$$|\Psi(\tau)\rangle = e^{-(\hat{H}-E_T)\tau} |\Psi(0)\rangle = \sum_i c_i e^{-(E_i-E_T)\tau} |\phi_i\rangle .$$  \hfill (70)

States with $E_i > E_T$ will decay. The projection operator is Green’s function

$$\hat{G}(\tau) = e^{-(\hat{H}-E_T)\tau} ,$$  \hfill (71)

satisfies the equation

$$-\frac{\partial \hat{G}(\tau)}{\partial \tau} = (\hat{H} - E_T)\hat{G} .$$  \hfill (72)

Represented by a coordinate-space matrix with elements

$$\langle R'|\hat{G}(\tau)|R\rangle = \langle R'| e^{-(\hat{H}-E_T)\tau} |R\rangle \equiv G(R',R;\tau) .$$  \hfill (73)
Short time $G(\tau)$: Operator splitting

Usually $\hat H = \hat T + \hat V$, and $[\hat T, \hat V] \neq 0 \Rightarrow$ we don't know $e^{-(\hat H - E\tau)}$ exactly. But we can approximate,

$$G(\tau) = e^{-(\hat H - E\tau)\tau} = e^{-(\hat T + \hat V - E\tau)\tau} = \underbrace{e^{-\hat T\tau}}_{G_{\hat T}(\tau)} \underbrace{e^{-(\hat V - E\tau)\tau}}_{G_{\hat V}(\tau)} + O(\tau^2). \quad (74)$$

The potential part is easy,

$$G_{\hat V}(R', R; \tau) = e^{-(V(R) - E\tau)\tau} \delta(R' - R). \quad (75)$$

The kinetic part is a bit trickier (to be derived later),

$$G_{\hat T}(R', R; \tau) = (4\pi\lambda\tau)^{-3N/2} e^{-\frac{(R-R')^2}{4\lambda\tau}}. \quad (76)$$

Put together, we get the "primitive approximation",

$$G(R', R; \tau) \approx (4\pi\lambda\tau)^{-3N/2} e^{-\frac{(R-R')^2}{4\lambda\tau}} e^{-(V(R) - E\tau)\tau}. \quad (77)$$
Diffusion $G_\hat{T}(\tau)$

Let’s derive the coordinate-space representation of the diffusion Green’s function. We use the eigenstate expansion ($K$ is a $3N$-dimensional momentum space vector),

$$\hat{T} \phi_K(R) = (-\lambda \nabla_R^2) \phi_K(R) = \lambda K^2 \phi_K(R) \tag{78}$$

$$\phi_K(R) = \langle R|K \rangle = \frac{1}{\sqrt{L^{3N}}} e^{-iK \cdot R} \tag{79},$$

and the states $|K\rangle$ are a complete basis. The coordinate-space representation of $G_\hat{T}(\tau)$ is

$$G_\hat{T}(R', R; \tau) \equiv \langle R'| e^{-\tau \hat{T}} | R \rangle = \sum_K \langle R'| e^{-\tau \hat{T}} | K \rangle \langle K | R \rangle \tag{80}$$

$$= \sum_K \langle R'| e^{-\tau \lambda K^2} | K \rangle \phi_K^*(R) = \sum_K \phi_K(R') \phi_K^*(R) e^{-\tau \lambda K^2} \tag{81}$$

$$= \frac{1}{L^{3N}} \sum_K e^{-iK \cdot (R' - R)} e^{-\tau \lambda K^2}. \tag{82}$$
Diffusion $G_T(\tau)$

The momentum values are $2\pi/L$ apart, and for large $L$ the sum can be converted to an integral.\(^5\)

$$\sum_K \approx \frac{L^{3N}}{(2\pi)^{3N}} \int dK$$

$$\frac{1}{L^{3N}} \sum_K e^{-iK \cdot (R' - R)} e^{-\tau \lambda K^2} \approx \frac{1}{(2\pi)^{3N}} \int dK e^{-iK \cdot (R' - R)} e^{-\tau \lambda K^2},$$

where the integral is a Fourier transform of a Gaussian, leading to the nice result

$$G_T(R', R; \tau) = \left(4\pi \lambda \tau\right)^{-3N/2} e^{-\frac{(R-R')^2}{4\lambda \tau}}.$$\(^{(85)}\)

This is usually an excellent approximation, so I left out the ”$\approx$”.

\(^5\)If not, evaluate the sum.
Simulation of $G_T(\tau)$: Diffusion

We want to compute evolution $|\Psi(\tau)\rangle = G_T(\tau)|\Psi(0)\rangle$, which is

$$\psi(R'; \tau) = \int dR G_T(R', R; \tau) \psi(R; 0) \approx \frac{1}{M} \sum_{i=1}^{M} \psi(R_i; 0), \quad (86)$$

where $R_i$ is sampled from $G_T(R', R; \tau)$ (fixed $R'$ and $\tau$).

$G_T(R', R; \tau)$ is a product of $3N$ Gaussians (in 3D), so all of them can be sampled independently $\Rightarrow 3N$ independent diffusions – but what is $\sigma$?

Compare the exponents:

$$2\sigma^2 = 4\lambda \tau \Rightarrow \sigma = \sqrt{2\lambda \tau} . \quad (87)$$

Result: we can sample points $R_i$ by a random walk with the rule

$$R_{i+1} = R_i + \sqrt{2\lambda \tau} \eta , \quad (88)$$

where $\eta = (\eta_x, \eta_y, \eta_z)$, $\eta_{x,y,z}$ sample standard normal distribution.
MC representation of the many-body wave function

In VMC we had an analytical wave function, but in DMC we don’t have an analytical form for \( \Psi(R, \tau) \). Instead, DMC represents \( \Psi(R, \tau) \) in terms of a finite number of samples \( \{R\} \). These configurations are made of “walkers” (they walk a Markov chain).

Consequences:

1. Need quite a large sample \( \{R\} \) to represent \( \Psi(R, \tau) \)
   - Use of the order of 100–10000 (depending on the problem) configurations \( R \), each is one possible configuration in \( \Psi(R, \tau) \)
   - Slower than VMC by about the number of configurations
   - Many walkers \( \Rightarrow \) consumes memory
   - Each configuration is independent \( \Rightarrow \) very good parallelization! Only now and then collect data and do branching.

2. Too few configurations means \( \Psi(R, \tau) \) is poorly represented.
   \( \Rightarrow \) The algorithm must keep the number of configurations large – but not too large, or you’re out of memory.
First DMC algorithm

Pick a large number of configurations \( \{ \mathbf{R} \} \) to begin with.

1. **Diffusion** due to \( G_{\hat{T}}(\tau) \): Move from \( \mathbf{R} \rightarrow \mathbf{R}' \) diffusively, in other words move each walker according to

\[
    r'_i = r_i + \eta \sqrt{2\lambda \tau},
\]

where \( \eta \) is a d-dimensional Gaussian random variable with zero mean and unit variance.

2. **Branching** due to \( G_{\hat{V}}(\tau) \): The wave function \( \Psi(\mathbf{R}, 0) \) is multiplied by the factor \( e^{-(V(\mathbf{R}) - E_T)\tau} \). So we calculate \( V(\mathbf{R}) \) for each configuration and see whether this factor is more or less than one, and create/destroy configurations accordingly. Walkers are not moving.

Alas, this is no good if the potential is singular: diffusion takes walkers sooner or later to a near-singular position.
First DMC algorithm

The factor $e^{-(V(R) - E_T)\tau}$ makes the algorithm problematic:

- **Repulsive potential**, $V(R) \gg 0$ if particles come too close. The walker is killed in branching. Easily lots of killed walkers $\Rightarrow$ only a few may survive $\Rightarrow$ bad statistics and large fluctuations in number of walkers.

- **Attractive potential**, $V(R) \ll 0$, if particles come too close. The walker branches to very many walkers. $\Rightarrow$ large jump up in number of walkers $\Rightarrow$ bad statistics and large fluctuations in number of walkers.

Bad statistics results if present walkers are descendants of too few walkers.

Conclusion: we’d better improve the algorithm so that walkers stay away from regions of large $|V(R)|$. This can be done with **importance sampling**.
Importance sampling in DMC

Solve the imaginary-time evolution of the product

\[ f(R; \tau) \equiv \Psi(R; \tau)\varphi_T(R), \quad (89) \]

knowing that the evolution of \( \Psi(R; \tau) \) is given by the imaginary-time Schrödinger equation,

\[-\frac{\partial \Psi(R; \tau)}{\partial \tau} = (\hat{\mathcal{H}} - E_T)\Psi(R; \tau). \quad (90)\]

Insert \( \Psi(R; \tau) = f(R; \tau)/\varphi_T(R) \) to get

\[-\frac{\partial f(R; \tau)}{\partial \tau} = -\lambda \nabla_R^2 f(R; \tau) + \lambda \nabla_R \cdot [F(R)f(R; \tau)] + [E_L(R) - E_T] f(R; \tau). \quad (91)\]

**drift force (or drift velocity, quantum force)**

\[ F(R) \equiv \nabla_R \ln |\varphi_T(R)|^2 = 2 \frac{\nabla_R \varphi_T(R)}{\varphi_T(R)}, \quad (92) \]

and the familiar **local energy** \( E_L(R) \equiv \varphi_T(R)^{-1}\hat{\mathcal{H}}\varphi_T(R) \).
Green’s function for importance sampling DMC

The imaginary-time evolution is

\[-\frac{\partial f(R; \tau)}{\partial \tau} = -\lambda \nabla^2_R f(R; \tau) + \lambda \nabla_R \cdot [F(R)f(R; \tau)] + [E_L(R) - E_T] f(R; \tau) \]

\[\equiv \hat{\mathcal{H}}_F f(R; \tau),\]  \(\text{(93)}\)

which defines a new operator, \(\hat{\mathcal{H}}_F\). Again the formal solution is

\[f(R; \tau) = e^{-\tau \hat{\mathcal{H}}_F} f(R; 0),\]  \(\text{(94)}\)

and the Green’s function is again

\[G(R, R'; \tau) = \langle R' | e^{-\tau \hat{\mathcal{H}}_F} | R \rangle.\]  \(\text{(95)}\)

Once we know what this is, we can write the evolution as an integral equation,

\[f(R; \tau) = \int dR' G(R, R'; \tau)f(R'; 0).\]  \(\text{(96)}\)
Green's function for importance sampling DMC

Without importance sampling we did two things:

1. Decide how accurate Green's function we want. We settled for the most crude Trotter expansion,

\[
e^{-\tau \hat{H}} \approx e^{-\tau \hat{T}} e^{-\tau \hat{V}},
\]

which has timestep error \(O(\tau^2)\).

2. Solve \(G_{\hat{T}}(\mathbf{R}', \mathbf{R}; \tau) = \langle \mathbf{R}' | e^{-\tau \hat{T}} | \mathbf{R} \rangle \) and \(G_{\hat{V}}(\mathbf{R}', \mathbf{R}; \tau) = \langle \mathbf{R}' | e^{-\tau \hat{V}} | \mathbf{R} \rangle\)

And we were done. For better accuracy we could improve the Trotter expansion. Importance sampling adds one complication, namely the drift term

\[
\hat{H}_F f(\mathbf{R}; \tau) = -\lambda \nabla^2_{\mathbf{R}} f(\mathbf{R}; \tau) + \lambda \nabla_{\mathbf{R}} \cdot [\mathbf{F}(\mathbf{R}) f(\mathbf{R}; \tau)] + [E_L(\mathbf{R}) - E_T] f(\mathbf{R}; \tau)
\]

\[
\equiv \hat{T}_F f(\mathbf{R}; \tau) + [E_L(\mathbf{R}) - E_T] f(\mathbf{R}; \tau).
\]

Evolution with only diffusion and drift is known as the Fokker-Planck equation.
Fokker-Planck operator

The drift-diffusion a.k.a Fokker-Planck evolution is

\[- \frac{\partial f(\mathbf{R}; \tau)}{\partial \tau} = -\lambda \nabla^2_R f(\mathbf{R}; \tau) + \lambda \nabla_R \cdot [\mathbf{F}(\mathbf{R})f(\mathbf{R}; \tau)] \equiv \hat{T}_F f(\mathbf{R}; \tau) \]  \hspace{1cm} (99)

Define the drift operator,

\[\mathbf{F}(\hat{\mathbf{R}}) | \mathbf{R} \rangle \equiv \mathbf{F}(\mathbf{R}) | \mathbf{R} \rangle . \]  \hspace{1cm} (100)

to get

\[\nabla_R \cdot [\mathbf{F}(\mathbf{R})f(\mathbf{R}; \tau)] = \nabla_R \cdot \left[ \langle \mathbf{R} | \mathbf{F}(\hat{\mathbf{R}}) | f(\tau) \rangle \right] = \nabla_R \cdot \mathbf{F}(\hat{\mathbf{R}})f(\mathbf{R}; \tau) . \]  \hspace{1cm} (101)

so the Fokker-Planck operator is

\[\hat{T}_F = -\lambda \nabla^2_R + \lambda \nabla_R \cdot \mathbf{F}(\hat{\mathbf{R}}) . \]  \hspace{1cm} (102)
Fokker-Planck operator

The Fokker-Planck operator is usually written in terms of the momentum operator $\hat{P} = -i\nabla_R$,

$$\hat{T}_F = \lambda \hat{P}^2 + i \lambda \hat{P} \cdot F(\hat{R}) .$$  \hspace{1cm} (103)

With timestep error $O(\tau^2)$,

$$e^{-\tau \hat{H}_F} \approx e^{-\tau \hat{T}_F} e^{-\tau \hat{E}_L - E_T}$$  \hspace{1cm} (104)

where, for shortness, I defined ”local energy operator” $\hat{E}_L$ similar to the potential operator $\langle R'|\hat{E}_L|R \rangle = E_L(R)\delta(R' - R)$. That’s simple, the Fokker-Planck Green’s function is more involved. We need to evaluate

$$G_{T_F}(R', R; \tau) \equiv \langle R'|e^{-\tau \hat{T}_F}|R \rangle .$$  \hspace{1cm} (105)

But before we do, let’s look at why did we ”give up” and defined a drift operator.
Fokker-Planck evolution

Why didn’t we solve the drift-diffusion matrix elements and be happy? Try to proceed as usual, and turn our attention to the eigenfunction expansion – just to show you how and why this is \textit{wrong}. Let’s boldly define

\[
\hat{T}_F Q_Q(R) = q_Q Q_Q(R) \quad (\textit{not valid !})
\]

\[
Q_Q(R) = \langle R|Q\rangle = \sum_K \langle R|K\rangle\langle K|Q\rangle = \sum_K a_{Q,K} \phi_K(R) .
\]

(106)

(107)

Repeat the steps with free diffusion,

\[
G_{TF}(R', R; \tau) = \sum_Q e^{-\tau q_Q} Q_Q^*(R') Q_Q(R) \quad (\textit{not valid !}) .
\]

(108)

and we are done. But hold on, \textit{can} we solve the eigenproblem for \(\hat{T}_F\)?
Fokker-Planck evolution

Let’s try to solve the Fokker-Planck eigenproblem (non-existent, but we were curious). First,

\[
\hat{T}_F \phi_K(R) = -\lambda \nabla^2_R \phi_K(R) + \lambda \nabla_R \cdot [F(R) \phi_K(R)] .
\] (109)

Insert the eigenstates of \( \hat{T} \), \( \phi_K(R) = \frac{1}{\sqrt{L^{3N}}} e^{-iK \cdot R} \),

\[
\hat{T}_F \phi_K(R) = \lambda K^2 \phi_K(R) + \lambda \nabla_R \cdot [F(R) \phi_K(R)] .
\] (110)

Using \( \nabla_R \phi_K(R) = -iK \phi_K(R) \) we get

\[
\lambda \nabla_R \cdot [F(R) \phi_K(R)] = \lambda [\nabla_R \cdot F(R)] \phi_K(R) + i\lambda K \cdot F(R) \phi_K(R) .
\] (111)
Fokker-Planck evolution

So far we have

$$\hat{T}_F \phi_K(R) = \left[ \lambda K^2 + \lambda [\nabla_R \cdot \mathbf{F}(R)] + i \lambda K \cdot \mathbf{F}(R) \right] \phi_K(R).$$ \hspace{1cm} (112)

The quantity in brackets depends on \( R \), so it’s not an eigenvalue. The imaginary unit gives us a hint (you can prove it also rigorously) that \( \hat{T}_F \) is non-hermitian, so we conclude that

**The Fokker-Planck operator \( \hat{H}_F \) is non-hermitian** \(^6\),

So what’s the problem? The eigenfunction expansion in Eq. (108) is valid only for hermitian operators, non-hermitian operators are much harder to deal with. Another fine mess importance sampling got us into.

Let’s return to the drawing board and try another route to solve the Fokker-Planck (drift-diffusion) problem.

\(^6\)Non-self-adjoint, to be more precise.
Fokker-Planck evolution – again

Let’s make a new start and look again at the Fokker-Planck equation,

\[- \frac{\partial f(R; \tau)}{\partial \tau} = -\lambda \nabla^2_R f(R; \tau) + \lambda \nabla_R \cdot [F(R)f(R; \tau)], \quad (113)\]

Instead of working with the evolution of the probability density \( f(R; \tau) \), we can write an equivalent stochastic PDE for time \textit{evolution of the observable}, \( R(\tau) \). The equation for \( R(\tau) \) is a \textit{Langevin equation}. \(^7\)

The basic problem is obvious: diffusion and drift don’t commute. We can try splitting them to two separate evolution processes we can handle. Be careful, you’d want the same accuracy in \( \tau \) as you had in the original operator splitting. A better accuracy may be too costly and a worse accuracy sets the overall accuracy to lower. If we are happy with \( \mathcal{O}(\tau^2) \) accuracy, then \( G_F(\tau) = e^{-\tau(-\lambda \nabla^2_R + \lambda \nabla_R \cdot F(\hat{R}))} \approx e^{-\tau \hat{T}} e^{-\tau \lambda \nabla_R \cdot F(\hat{R})} \) is good enough also here. The first part is diffusion, the second is drift.

\(^7\)Some mathematicians dislike Langevin equations because of the so-called Itô-Stratonovich dilemma: there are two valid interpretations of how certain stochastic integrals should be evaluated; the values of the integrals disagree.
Drift evolution

Concentrate only on solving only the drift evolution,

\[- \frac{\partial f(R; \tau)}{\partial \tau} = \lambda \nabla_R \cdot [F(R)f(R; \tau)] \text{ (only drift)} \] (114)

The drift Green’s function is propagation of particles along a trajectory,

\[\langle R' | e^{-\tau \lambda \nabla_R \cdot F(\hat{R})} | R \rangle = \delta(R' - R(\tau)) \] (115)

so let’s ask

**What are the trajectories R(\tau) of particles under drift?**

The drift differential equation (114) is a *continuity equation*, usually written for density \(\rho(r)\) and current \(j(r)\) as \(\partial \rho / \partial t = \nabla \cdot j\). We can interpret the function \(f(R; \tau)\) as density and \(\lambda F(R)f(R; \tau)\) as current, so we can pick the solution from literature: the trajectories are solutions of

\[\frac{dR(\tau)}{d\tau} = \lambda F(R(\tau)) \] (116)

The solution that is good enough for present purposes is

\[R(\tau) = R(0) + F(R(0)) \lambda \tau + O(\tau^2)\] (117)
The Langevin equation of Fokker-Planck evolution

Now we know what drift does (correct to $O(\tau^2)$),

$$\langle R'| e^{-\tau \lambda \nabla_R \cdot F(\hat{R})} | R \rangle = \delta(R' - R - \lambda \tau F(R)) ,$$  \hspace{1cm} (118)

A random walk with both diffusion and drift is simply

$$R' = R + \sqrt{2 \lambda \tau} \eta + \lambda \tau F(R) .$$  \hspace{1cm} (119)

This is the approximate solution of the stochastic Langevin equation, which was equivalent to the Fokker-Planck equation for the probability density $f(R; \tau)$.\(^8\)

This approximate solution begins to fails because the larger the timestep, the more frequently diffusion takes particles to very large drift. This error may not be healed in subsequent moves and you begin to get biased results.

\(^8\)It’s sometimes stated that the Langevin equation is analogous to Heisenberg picture, Fokker-Planck equation is analogous to Schrödinger picture. Frankly, I don’t see the analogy.
A working first order DMC algorithm

1. Initialise many (≈ 100 – 500) configurations, where walkers are distributed according to $|\varphi_T(R)|^2$. We don’t want them all destroyed in the branching process, so we need to generate as diverse a set as possible. You can pick from a VMC run that samples $|\varphi_T(R)|^2$ positions $R$ now and then to a zeroth generation of configurations. The total number of configurations in a generation is its population.

2. In each configuration of the present generation, move the $i^{th}$ walker with

$$r_i^{\text{new}} = r_i + \lambda \tau F(R) + \eta \sqrt{2\lambda \tau}, \quad (120)$$

where $\eta$ is a 3-dimensional Gaussian random vector.

3. Calculate a weight for the move

$$\text{ratio} = \frac{|\varphi_T(R')|^2 G(R', R; \tau)}{|\varphi_T(R)|^2 G(R, R', \tau)}, \quad (121)$$

and accept the move $R \rightarrow R'$ with probability $\min[1, \text{ratio}]$ ("Metropolis question")
A working first order DMC algorithm

4. Take each configuration in turn and make $N_{\text{copies}}$ copies of it,

$$N_{\text{copies}} = \text{int} \left[ e^{-\tau(E_L(R)-E_T)} + \xi \right] \quad \xi \in U[0, 1], \quad (122)$$

5. Estimate the ground-state energy $E_0$ by taking the average of the local energies of current generation.

6. Keep the population healthy by adjusting the trial energy $E_T$. One way is to use a feedback mechanism,

$$E_T = E_0 + \kappa \ln \frac{N_{C_{\text{target}}}}{N_C}, \quad (123)$$

where $E_0$ is the best guess for the ground-state energy we have so far, $N_{C_{\text{target}}}$ is the population we’d like to have, and $N_C$ is the one we have now. The positive parameter $\kappa$ should have the smallest value that does the job.

7. go back to step 2
A working first order DMC algorithm

Comments:

• The accept/reject test is a Metropolis-Hastings test.
  
  – The Hamiltonian with drift is *non-hermitian*; the accept/reject step ensures that the detailed balance condition is satisfied ⇒ makes sure the points $\mathbf{R}$ are really sampled from the distribution $f(\mathbf{R}; \tau) = \Psi(\mathbf{R}; \tau) \varphi_T(\mathbf{R})$.
  
  – The acceptance *must be* $> 99.9 \%$ – if not, you time step is too large. At your own discretion you may even omit the test and assume that the acceptance is very high anyhow, but then you are on your own.

• Controlling the number of configurations (sets of walkers) should be done with kid gloves. The more you force it, the more fluctuations increase elsewhere. No matter how you control the population you will be upsetting randomness, so be gentle.

• Use as large a population as you can afford. Something like 5000 configurations is common. Debugging can use a smaller set.
Higher order DMC

The order of DMC comes from how we deal with non-commuting parts in the Hamiltonian $\hat{H}$. With importance sampling, we must write $e^{-\tau \hat{H}_F}$ in terms of simpler, albeit approximate, matrix elements:

1. **First operator splitting**: $\hat{H}_F = \hat{T}_F + \hat{E}_L$, $[\hat{T}_F, \hat{E}_L] \neq 0$
   Separate the Fokker-Planck (drift-diffusion) operator $\hat{T}_F$ and the local energy operator $\hat{E}_L$.

2. **Second operator splitting**: $\hat{T}_F = \hat{T} + \lambda \nabla_R \cdot \mathbf{F}(\mathbf{R})$, $[\hat{T}, \lambda \nabla_R \cdot \mathbf{F}(\mathbf{R})] \neq 0$
   The Fokker-Planck equation describing probability density evolution in diffusion under drift is cast to a Langevin equation, an equation of motion of a single walker under a velocity field.

To get second or higher order DMC we must improve both splittings in unison, and solve the Langevin equation without loosing accuracy.
What Hamiltonian did we actually use?

Let's ask a question: What "fake Hamiltonian" $\hat{H}'$ has the exact splitting \(^9\)

$$e^{-\tau \hat{H}'} = e^{-\tau \hat{T}} e^{-\tau \hat{V}} \ ?$$

(124)

The Campbell-Baker-Hausdorff (C-B-H) formula \(^10\) states that

$$e^{\hat{A}} e^{\hat{B}} = e^{\hat{A}+\hat{B}+\frac{1}{2}[\hat{A},\hat{B}]+\frac{1}{12}[\hat{A}-\hat{B},[\hat{A},\hat{B}]]+...} \ ,$$

(125)

so (drop terms beyond $\tau^3$)

$$e^{\tau \hat{T}} e^{\tau \hat{V}} = e^{\tau (\hat{H}+\frac{1}{2} \tau [\hat{T},\hat{V}]+\frac{1}{12} \tau^2 [\hat{T}-\hat{V},[\hat{T},\hat{V}]]}) \equiv e^{\tau \hat{H}'},$$

(126)

and read the operator (use $[\hat{T},\hat{V}] = [\hat{H}-\hat{V},\hat{V}] = [\hat{H},\hat{V}]$),

$$\hat{H}' = \hat{H} + \frac{1}{2} \tau [\hat{H},\hat{V}] - \frac{1}{12} \tau^2 [\hat{H}-2\hat{V},[\hat{H},\hat{V}]].$$

(127)

---


\(^10\)An old theorem (1890,1902), has to do with Lie algebra of a Lie group.
Energy accuracy without importance sampling

Now we can find out how bad mistake we did in energy. After DMC
without importance sampling has converged, we get the energy

\[ \langle E \rangle \equiv \langle E_L \rangle_{\phi_0} = \langle \phi_0 | \hat{H}' | \phi_0 \rangle \]  

= \[ E_0 + \frac{1}{2} \tau \langle \phi_0 | [\hat{H}, \hat{V}] | \phi_0 \rangle - \frac{1}{12} \tau^2 \langle \phi_0 | [\hat{H} - 2\hat{V}, [\hat{H}, \hat{V}]] | \phi_0 \rangle \]  

= \[ E_0 - \frac{1}{12} \tau^2 \langle \phi_0 | [\hat{H} - 2\hat{V}, [\hat{H}, \hat{V}]] | \phi_0 \rangle , \]  

(128)

(129)

(130)

where the linear term is zero, because \( |\phi_0\rangle \) is an eigenstate of \( \hat{H} \),

\[ \langle \phi_0 | [\hat{H}, \hat{V}] | \phi_0 \rangle = \langle \phi_0 | \hat{H} \hat{V} - \hat{V} \hat{H} | \phi_0 \rangle = 0 . \]  

(131)

**Without importance sampling the energy converges as \( \tau^2 \)**

**with**

\[ e^{-\tau \hat{H}} \approx e^{-\tau \hat{T}} e^{-\tau \hat{V}} \]
Energy accuracy with importance sampling

Importance sampling makes DMC sample the product $\phi_0(R)\varphi_T(R)$, and we evaluate the mixed estimator

$$
\langle E \rangle \equiv \langle E_L \rangle_{\phi_0\varphi_T} = \langle \varphi_T | \hat{H}^\prime | \phi_0 \rangle
$$

$$
= E_0 + \frac{1}{2} \tau \langle \varphi_T | [\hat{H}, \hat{V}] | \phi_0 \rangle - \frac{1}{12} \tau^2 \langle \varphi_T | [\hat{H} - 2\hat{V}, [\hat{H}, \hat{V}]] | \phi_0 \rangle,
$$

which converges only linearly, because $\langle \varphi_T | [\hat{H}, \hat{V}] | \phi_0 \rangle \neq 0$.

With importance sampling the energy converges as $\tau$ with

$$
e^{-\tau \hat{H}} \approx e^{-\tau \hat{T}} e^{-\tau \hat{V}}$$
Energy accuracy with importance sampling

We saw this linear convergence in the demo program.

![Graph of He atom energy with DMC and Exact calculations](image)

- DMC
- Exact

**Heatom (spins up-down)**

E (Hartree) vs. timestep (1/Hartree)

-2.9065 - 2.906
-2.9055 - 2.905
-2.9045 - 2.904
-2.9035 - 2.903

-2.90375 - 0.362546 timestep

Quantum Monte Carlo (QMC)
Propagators with second order accuracy

It’s simple to construct a 2nd order splitting, one popular choice is

\[ e^{-\tau \hat{A} + \hat{B}} \approx e^{-\frac{1}{2} \tau \hat{A}} e^{-\tau \hat{B}} e^{-\frac{1}{2} \tau \hat{A}} \] (133)

Next we must also solve the drift trajectories from Eq. (116)

\[ \frac{dR(\tau)}{d\tau} = \lambda F(R(\tau)) \] (134)

to second order. The drift changes as particles move, so Eq. (117) could be improved to (2nd order Runge-Kutta)

\[ R(\tau) = R(0) + \lambda \tau F(R(0)) + \frac{1}{2} \lambda \tau F(R(0)) + \mathcal{O}(\tau^3). \] (135)

There are several alternatives (see Chin), which differ in how many drift evaluations (expensive) are needed and what is the range of quadratic behaviour. I won’t go into details, but push forward to 4th order algorithms.
Propagators with fourth order accuracy

Suzuki: there is no 4th order splitting

\[ e^{-\tau(\hat{A}+\hat{B})} = \prod_{i=1}^{N} e^{-\tau a_i \hat{A}} e^{-\tau b_i \hat{B}} \]  \hspace{1cm} (136)

without some \( a_i < 0 \) and \( b_i < 0 \). \(^{11}\)

This we cannot have in DMC, because a negative \( t_i \) means \( \langle R' | e^{-\tau t_i \hat{T}} | R \rangle \) is inverse diffusion (backward in time), which cannot be simulated.

Something has to be added, and the that's some commutator(s) of \( \hat{T} \) and \( \hat{V} \). A convenient one to keep is related to the classical force on particle \( i \), \(^{12}\)

\[ [\hat{V}, [\hat{T}, \hat{V}]] = 2\lambda |\nabla_R V(R)|^2 = 2\lambda \sum_{i=1}^{N} |f_i(R)|^2, \quad f_i = -\nabla_i V. \]  \hspace{1cm} (137)


\(^{12}\) The operator \([\hat{V}, [\hat{T}, \hat{V}]]\) acts on some \( \phi(R) \).
The commutator is evaluated like this, $\hat{T} = -\lambda \nabla^2 R$, $V := V(R)$,

$$
[\hat{V}, [\hat{T}, \hat{V}]] \phi(R) = \left[-V^2 \hat{T} - \hat{T} V^2 + 2V \hat{T} V\right] \phi \\
= \lambda \left[V^2 \nabla^2 \phi + \nabla^2 (V^2 \phi) - 2V \nabla^2 (V \phi)\right] \\
= \lambda \left[V^2 \nabla^2 \phi + \nabla \cdot [2V(\nabla V) \phi + V^2 \nabla \phi] \\
- 2V \nabla \cdot [(\nabla V) \phi + V \nabla \phi]\right] \\
= \lambda \left[V^2 \nabla^2 \phi + 2(\nabla V)^2 \phi + 2(V \nabla^2 V) \phi + 2V(\nabla V) \cdot \nabla \phi \\
+ 2V \nabla V \cdot \nabla \phi + V^2 \nabla^2 \phi \\
- 2V \nabla^2 V \phi - 2V \nabla V \cdot \nabla \phi - 2V \nabla V \cdot \nabla \phi - 2V^2 \nabla^2 \phi\right]
$$

coloured terms cancel, only 2nd term survives

$$
= 2\lambda (\nabla V)^2 \phi .
$$
Propagators with fourth order accuracy

Caleidoscope of 4th order methods (incomplete list):

- **Trotter** \(^{13}\) (divide \(\tau\) to infinitely many pieces)

\[
e^{-\tau \hat{H}} = \lim_{M \to \infty} \left[ e^{-\frac{\tau}{M} \hat{T}} e^{-\frac{\tau}{M} \hat{V}} \right]^M, \tag{141}
\]

- **Ruth-Forest** (Ruth (1983, private channels), Forest and Ruth (1990), re-discovered several times around 1990 by Campostrini and Rossi and by Candy and Rozmus) \(^{14}\)

- **Takahashi-Imada** (1984)\(^ {15}\) and Li-Broughton (1987)\(^ {16}\)

- **Chin** (1997)\(^ {17}\)

---


\(^{14}\) E. Forest and R. D. Ruth, Physica D. 43, 105 (1990)


Propagators with fourth order accuracy

Forest-Ruth

\[ \hat{G}_{\text{FR}}(\tau) = e^{-v_3 \tau \hat{\mathcal{V}}} e^{-t_3 \hat{T}} e^{-v_2 \tau \hat{\mathcal{V}}} e^{-t_2 \tau \hat{T}} e^{-v_1 \tau \hat{\mathcal{V}}} e^{-t_1 \tau \hat{T}} e^{-v_0 \tau \hat{\mathcal{V}}}, \]  

with \( s = 2^{1/3} = 1.25992, \)

\[ v_0 = v_3 = \frac{1}{2} \frac{1}{2 - s} \, (> 0) \quad , \quad v_1 = v_2 = -\frac{1}{2} \frac{s - 1}{2 - s} \, (> 0) \]  

\[ t_1 = t_3 = \frac{1}{2 - s} \, (> 0) \quad , \quad t_2 = -\frac{s}{2 - s} \, (< 0!) \]  

From right to left, the potential will be evaluated at times 0, \( t_1 \tau, (t_1 + t_2)\tau \, (< 0!), \) and \( \tau. \) Suzuki was right, there are negative coefficients. These make the application quite tricky. The good thing about FR is that it only needs 6 FFT's to propagate time \( \tau, \)
Propagators with fourth order accuracy

Takahashi-Imada  "not fully fourth order"

\[ \hat{G}_{T1}(\tau) = e^{-\tau \hat{T}} e^{-\tau \hat{V}} - \frac{1}{24} \tau^3 [\hat{V}, [\hat{T}, \hat{V}]] e^{-\tau \hat{T}} \]  \hspace{1cm} (145)

Chin  Several possible forms, one is

\[ \hat{G}_{Chin}(\tau) = e^{-\nu_0 \tau \hat{V}} e^{-t_1 \tau \hat{T}} e^{-\nu_1 \tau \hat{V}} e^{-t_2 \tau \hat{T}} e^{-\nu_1 \tau \hat{W}} e^{-t_1 \tau \hat{T}} e^{-\nu_0 \tau \hat{V}} \]  \hspace{1cm} (146)

\[ \hat{W} \equiv \hat{V} + \frac{u_0}{\nu_1} \tau^2 [\hat{V}, [\hat{T}, \hat{V}]] , \]  \hspace{1cm} (147)

\[ \nu_0 = \frac{6 t_1 (t_1 - 1) + 1}{12 (t_1 - 1) t_1} , \hspace{1cm} t_2 = 1 - 2 t_1 \]  \hspace{1cm} (148)

\[ \nu_1 = \frac{1}{2} - \nu_0 \hspace{1cm} u_0 = \frac{1}{48} \left( \frac{1}{6 t_1 (1 - t_1)^2} - 1 \right) , \]  \hspace{1cm} (149)

and \( t_1 \) is tunable, \( 0 \leq t_1 \leq 1 - 1/\sqrt{3} \). Even 5th order errors may cancel with \( t_1 = 0.35023 \).  \textit{All coefficients are positive.}
What can a good propagator do?

We have a good approximation for $\hat{G}(\tau) = e^{-\tau \hat{H}}$, such as

$$\hat{G}_{\text{Chin}}(\tau) = \hat{G}(\tau) + \mathcal{O}(\tau^5). \quad (150)$$

Nothing prevents us from overextending the validity, as long as we know that we are improving. Take a trial wave function $\varphi_T(R)$ and propagate it,

$$\langle R|\varphi_T(\tau_0)\rangle = \varphi_T(R; \tau_0) = \int dR' \langle R| G_{\text{Chin}}(\tau_0)|R'\rangle \varphi_T(R') . \quad (151)$$

We can use VMC to see how far in $\tau_0$ we can go until the energy begins to increase: find $\tau_0$ were

$$E(\tau_0) = \frac{\langle \varphi_T(\tau_0)|\hat{H}|\varphi_T(\tau_0)\rangle}{\langle \varphi_T(\tau_0)|\varphi_T(\tau_0)\rangle} \quad (152)$$

is at minimum (approximately).\(^{18}\)

\(^{18}\)In VMC the normalization is irrelevant.
What can a good propagator do?

The propagation can now be viewed as a multi-dimensional integral,

\[
\varphi_T(R; \tau_0) = \int dR' dR_1 dR_2 e^{-v_0 \tau_0 V(R)} \langle R | e^{-t_1 \tau_0 \hat{T}} | R_2 \rangle e^{-v_1 \tau_0 W(R_2)} \]

\[
\langle R_2 | e^{-t_2 \tau_0 \hat{T}} | R_1 \rangle e^{-v_1 \tau_0 W(R_1)} \langle R_1 | e^{-t_1 \tau_0 \hat{T}} | R' \rangle e^{-v_0 \tau_0 V(R')} \varphi_T(R').
\]

Notice that this applied one \( G_{\text{Chin}} \) for a single, long time step \( \tau_0 \).

---

---

\[19\] Rota et al. Physical Review E 81, 016707 (2010); The integrated coordinates can also be thought of as "shadow coordinates", as in the variational shadow wave function by L. Reatto, F. Pederiva et al.
What can a good propagator do?

Large $\tau$ means fast projection and less correlated points in DMC: 20

Liquid He$^4$: Timesteps 0.01 (dots), 0.05 (small points) and 0.2 (large points). Getting to the ground state takes 10 times more updates with a 2nd order propagator than with a 4th order one.
Fixed-node DMC

Fixed-node DMC (FN-DMC): use antisymmetric trial wave function,

\[
\varphi_T(\ldots r_i, \ldots, r_j \ldots) = -\varphi_T(\ldots r_j, \ldots, r_i \ldots).
\]

(154)

A common way to achieve this is to use

\[
\varphi_T(R) = e^{U(R)} \sum_k d_k D_k[\phi](R)
\]

(155)

- Jastrow factor \(e^U\)
  - particle-particle correlations
  - takes care of particle-particle potentials, especially singularities
- Slater determinants \(D_k\)
  - antisymmetrized single-particle orbitals \(\{\phi\}\)
  - takes care of Pauli principle
  - Functional of orbitals, evaluated at \(R\): in my clever notation \(D_k[\phi](R)\)
Jastrow factor

Often the potential doesn’t depend on symmetry, so we choose a convenient way to cancel it’s singularities. In electron systems, we have singular e-n and e-e potentials, so we need at least these correlations (indices $i, j$ for electrons, $\alpha$ for nuclei):

$$e^U(R) = e^{\sum_{i,\alpha} u_{en}(r_{i\alpha}) + \sum_{i<j} u_{ee}(r_{ij})} \text{ Jastrow factor .}$$ (156)

This can be supplemented with an e-e-n factor,

$$e^{\sum_{i<j,\alpha} u_{een}(r_{i\alpha}, r_{j\alpha}, r_{ij})},$$ (157)

which should be enough, because e-e-e-n is rare due to the Pauli principle.

*FN-DMC evolution optimizes the correlations to achieve lowest energy*
Slater determinants

For spin-1/2 fermions, one often uses two Slater determinants,

\[ \varphi_T(R) = e^{U(R)} D^\uparrow(r_1 \ldots r_N\uparrow) D^\downarrow(r_{N\uparrow+1} \ldots r_N) \, . \]  

Here the Slater determinant for \( N \) fermions with spins up is

\[ D^\uparrow(r_1 \ldots r_{N\uparrow}) = \frac{1}{\sqrt{N^\uparrow!}} \begin{vmatrix} \phi_1(r_1) & \phi_2(r_1) & \ldots & \phi_{N\uparrow}(r_1) \\ \phi_1(r_2) & \phi_2(r_2) & \ldots & \phi_{N\uparrow}(r_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_1(r_{N\uparrow}) & \phi_2(r_{N\uparrow}) & \ldots & \phi_{N}(r_{N\uparrow}) \end{vmatrix} . \]  

The single-particle orbitals \( \{ \phi \} \) can be taken from

- Hartree-Fock (HF) ← more popular
- Density functional theory (DFT)

HF has no correlations. This is presumably a good thing, because DMC has the liberty to optimize correlations in the Jastrow factor. DFT tries to add correlations to orbitals, which, presumably, prevents DMC to do its best in optimizing correlations.

\[ \text{While not antisymmetric under exchange of spin-up and spin-down fermions, the expectation values of spin-independent operators are valid. See W. M. C. Foulkes, L. Mitas, R. J. Needs and G. Rajagopal, Rev. Mod. Phys. 73, 33 (2001).} \]
Slater-type orbitals (STO’s)

First, hydrogen-type orbitals are not very useful. The complete set contains discrete status plus the continuum, hence $\infty$ number of states. Too many of them are needed in representation of core electrons. STO’s have the benefit of having only angular node, $R_{nl}$ is nodeless:

$$\phi_{nlm}(r) = R_{nl}(r) Y_{lm}(\theta, \phi)$$  \hspace{1cm} (160)

$$R_{nl}(r) = \frac{(2\zeta)^{3/2}}{\Gamma(2n + 1)} (2\zeta r)^{n-1} e^{-\zeta r},$$  \hspace{1cm} (161)

with a nuclear-charge-related constant $\zeta$. The notation 1s, 2s, 2p, 3d... is re-used. For more flexibility, the exponent can be chosen to depend on $nl$, $\zeta_{nl}$.
Gaussian-type orbitals (GTO’s)

The spherical harmonic GTO’s are chosen for their computational merits, rather than physical properties:

\[
\phi_{nlm}(r) = R_{nl}(r) Y_{lm}(\theta, \phi) \tag{162}
\]

\[
R_{nl}(r) = \frac{2(2\alpha)^{3/4}}{\pi^{1/3}} \sqrt{\frac{2^{2n-l-2}}{(4n-2l-3)!!}} \left(\sqrt{2\alpha r}\right)^{2n-l-2} e^{-\alpha r^2}. \tag{163}
\]

Here \(n!!\) is a special double factorial,

\[
n!! = \begin{cases} 
1 & n = 0 \\
n(n-2)(n-4)\ldots2 & n > 0, \text{even} \\
n(n-2)(n-4)\ldots1 & n > 0, \text{odd} \\
\frac{1}{(n+2)(n+4)\ldots1} & n < 0
\end{cases} \tag{164}
\]

Typically, a lot more GTO’s than STO’s are needed, but GTO’s are fast to evaluate. GTO’s have zero cusp and they die out faster in \(r\) than STO’s.
Cartesian forms

In cartesian coordinates Slater-type orbitals are

$$\phi_{ijk}^{STO}(x, y, z) = x^i y^j z^k e^{-\zeta r}, \quad (165)$$

and one often uses them as sums of Gaussian primitives,

$$\chi_n(x, y, z) = N_{ijk}^n x^i y^j z^k e^{-\alpha_n r^2}, \quad (166)$$

where the total angular momentum is $l = i + j + k$, resulting in one $s$-function ($l = 0$), three $p$-functions ($l = 1$), six $d$-functions ($l = 2$) etc. Simple ones are STO-nG, n Gaussians fitted to make an STO. Improvements are double-zeta (the $\zeta$’s in the exponent), two basis functions per atomic orbital, triple-zeta etc. Example 3: The ground state of Carbon is $1s^22s^22p^2$, and the orbitals $1s, 2s, 2p_x, 2p_y, 2p_z$ need minimal basis of 5 basis functions, or 10 double-zeta basis functions.
Slater-Condon rules

Notation: $|\Psi_{mn}^{pq}\rangle$ means from $\Psi$, change orbital $m$ to $p$ and $n$ to $q$, $(x_i := (r_i, \sigma_i))$

$$|\Psi\rangle := A[\varphi_1(x_1)\varphi_2(x_2)\ldots\varphi_m(x_m)\varphi_n(x_n)\ldots\varphi_N(x_N)]$$ (167)

$$|\Psi^p_m\rangle := A[\varphi_1(x_1)\varphi_2(x_2)\ldots\varphi_p(x_m)\varphi_n(x_n)\ldots\varphi_N(x_N)]$$ (168)

$$|\Psi_{mn}^{pq}\rangle := A[\varphi_1(x_1)\varphi_2(x_2)\ldots\varphi_p(x_m)\varphi_q(x_n)\ldots\varphi_N(x_N)].$$ (169)

The Slater-Condon rules for a one-body operator $\hat{F} = \sum_{i=1}^{N} f_i$ say, that the expectation value and matrix elements are

$$\langle \Psi | F | \Psi \rangle = \sum_{i=1}^{N} \langle \varphi_i | f | \varphi_i \rangle$$ (170)

$$\langle \Psi | F | \Psi^p_m \rangle = \langle \varphi_m | f | \varphi_p \rangle$$ (171)

$$\langle \Psi | F | \Psi_{mn}^{pq} \rangle = 0.$$ (172)

For example, the one-body Hamiltonian

$$\hat{H}_1 = \hat{T} + \hat{V}_1 = \sum_{i=1}^{N} (-\nabla_i^2 + V(r_i))$$ can change at most one orbital at a time.
Fixed-node DMC

The exact many-body ground state is a superposition of all possible Slater determinants. Say we have $N$ electrons, scattered on the available orbitals. One way to do this is (collective coordinates $x_i := (r_i, \sigma_i)$, antisymmetrization $A$)

- 1st term: all electrons on the lowest HF orbitals

  $$|\psi\rangle = A[\phi_1(x_1)\phi_2(x_2)\ldots\phi_m(x_m)\phi_n(x_n)\ldots\phi_N(x_N)]$$

- 2nd term: one electron excited to higher orbital $p$,

  $$|\psi^p_m\rangle = A[\varphi_1(x_1)\varphi_2(x_2)\ldots\varphi_p(x_m)\varphi_n(x_n)\ldots\varphi_N(x_N)]$$

- 3rd term: ...

HF takes a single Slater determinant (here the 1st term), and optimizes $\phi$’s for minimum energy

$\Rightarrow$ HF orbitals $\{\phi\}$ (solve a non-linear, integro-differential equation)

$\Rightarrow$ the HF energy – a good start.
Hartree-Fock + DMC

HF equations are (simplified to some extent)

\[
\left(\frac{\hbar^2}{2m} \nabla^2 + V(r) + V_H(r)\right) \phi_i(r) - q^2 \int d\mathbf{r'} \frac{\gamma(r, r')}{|\mathbf{r} - \mathbf{r'}|} \phi_i(r) = \epsilon_i \phi_i(r), \quad (173)
\]

where

\[
\gamma(r, r') \equiv \sum_j \phi_j^*(r) \phi_j(r') \quad (174)
\]

\[
V_H(\mathbf{R}) = q^2 \int d\mathbf{r'} \sum_j \frac{\phi_j^*(\mathbf{r'}) \phi_j(\mathbf{r'})}{|\mathbf{r} - \mathbf{r'}|} \quad \text{Hartree potential} \quad (175)
\]

Solution: Pick your favorite (constraint: money) from a long list of software.\textsuperscript{22} Mike Towler\textsuperscript{23} pinpoints the problem in feeding single-particle orbitals to a QMC code:

"Often people find that their HF/DFT code of choice is not supported, so they give up before they start."

\textsuperscript{22}wikidftsoftware
\textsuperscript{23}towler_pisa.pdf
Hartree-Fock + DMC

**HF energy** + **correlation energy** = exact energy (definition)

This much of the correlation energy is typically covered:

- Hartree-Fock: 0%
- Optimized Slater-Jastrow: ~85%
- FN-DMC: ~95%
- Exact: 100%

HF good ⇒ single-determinant Slater-Jastrow wave function OK

<table>
<thead>
<tr>
<th>atom</th>
<th>ground state</th>
<th>HF</th>
<th>Exp.</th>
<th>correlation</th>
</tr>
</thead>
<tbody>
<tr>
<td>He</td>
<td>1s(2)</td>
<td>-2.861679993</td>
<td>-2.90338683</td>
<td>-0.041706</td>
</tr>
<tr>
<td>Li</td>
<td>1s(2)2s(1)</td>
<td>-7.432726924</td>
<td>-7.478060323</td>
<td>-0.045333</td>
</tr>
<tr>
<td>Be</td>
<td>1s(2)2s(2)</td>
<td>-14.57302313</td>
<td>-14.6673564</td>
<td>-0.094333</td>
</tr>
<tr>
<td>B</td>
<td>1s(2)2s(2)2p(1)</td>
<td>-24.52906069</td>
<td>-24.541246</td>
<td>-0.012185</td>
</tr>
</tbody>
</table>
Dealing with Slater determinants

Marking $\varphi_T = e^U D^\uparrow D^\downarrow$, assuming $i \leq N^\uparrow$ (it not, swap $D^\downarrow$ and $D^\uparrow$),

$$\nabla_i \varphi_T = D^\uparrow D^\downarrow \nabla_i e^U + e^U D^\downarrow \nabla_i D^\uparrow$$  \hspace{1cm} (176)

$$F_i = 2 \frac{\nabla_i \varphi_T}{\varphi_T} = e^{-U} \nabla_i e^U + \frac{1}{D^\uparrow} \nabla_i D^\uparrow,$$  \hspace{1cm} (177)

and

$$\nabla_i^2 \varphi_T = D^\uparrow D^\downarrow \nabla_i^2 e^U + D^\downarrow \nabla_i D^\uparrow \cdot \nabla_i e^U + e^U D^\downarrow \nabla_i^2 D^\uparrow$$  \hspace{1cm} (178)

$$-\lambda \varphi_T^{-1} \nabla_i^2 \varphi_T = e^{-U} \nabla_i^2 e^U + \left[ \frac{1}{D^\uparrow} \nabla_i D^\uparrow \right] \cdot \left[ e^{-U} \nabla_i e^U \right] + \frac{1}{D^\uparrow} \nabla_i^2 D^\uparrow.$$  \hspace{1cm} (179)

Furthermore, in moving particles from old to new positions, we need the ratio

$$\frac{\Psi(R_{new})}{\Psi(R_{old})} = e^{U(R_{new}) - U(R_{old})} \frac{D^\uparrow(new)}{D^\uparrow(old)}.$$  \hspace{1cm} (180)
Dealing with Slater determinants
The Slater matrix $D$ of spin-up particles has components

$$D_{ji} = \phi_j(x_i) \quad i \leq N_{\uparrow}. \quad (181)$$

Remember the index order: *orbital*, *particle*.

- Changing the position of $i$th particle changes just the $i$th row of the Slater matrix:

$$D^{\uparrow}(\mathbf{r}_1...\mathbf{r}_{N_{\uparrow}}) = \frac{1}{\sqrt{N_{\uparrow}!}} \begin{vmatrix} \phi_1(\mathbf{r}_1) & \phi_2(\mathbf{r}_1) & \cdots & \phi_{N_{\uparrow}}(\mathbf{r}_1) \\ \cdots & \cdots & \cdots & \cdots \\ \phi_1(\mathbf{r}_i) & \phi_2(\mathbf{r}_i) & \cdots & \phi_{N_{\uparrow}}(\mathbf{r}_i) \\ \cdots & \cdots & \cdots & \cdots \\ \phi_1(\mathbf{r}_{N_{\uparrow}}) & \phi_2(\mathbf{r}_{N_{\uparrow}}) & \cdots & \phi_N(\mathbf{r}_{N_{\uparrow}}) \end{vmatrix}. \quad (182)$$

- The cofactor of matrix element $(i, j)$, $C_{ij}$, doesn’t depend on matrix elements on $i$th row, $j$th column.

$$C_{ij} = (-1)^{i+j} M_{ij}, \quad M_{ij} \text{ is the } i, j \text{ minor }, \quad (183)$$

and minor $M_{ij}$ leaves out matrix row $i$ and column $j$. 
Dealing with Slater determinants

If the $i$th particle moves, then

$$\bar{D} := (D^T)^{-1}$$

$$q_i := \frac{D(new)}{D(old)} = \sum_{j(\text{orb.})} \bar{D}_{ji}^{\text{old}} D_{ji}^{\text{new}} = \sum_{j(\text{orb.})} \bar{D}_{ji}^{\text{old}} \phi_j(r_i^{\text{new}}),$$

where $D = D^\uparrow$ or $D^\downarrow$. After an accepted move, update (Sherman–Morrison formula)

$$\bar{D}_{jk} = \begin{cases} 
\bar{D}_{jk}/q, & k = i \\
\bar{D}_{jk} - \frac{\bar{D}_{jk}}{q} \sum_{n(\text{orb.})} \frac{\bar{D}_{nk} \phi_n(r_i^{\text{new}})}{q} & k \neq i 
\end{cases}$$

The inverses $\bar{D}$ are computed in the beginning, and only updated as particles are moved.

Dealing with Slater determinants

Details:
Matrix inverse is related to the cofactor matrix,

$$\bar{D} := (D^T)^{-1} = \frac{C}{\det D} \Rightarrow D^{-1} = \frac{C^T}{\det D},$$  \hspace{1cm} (187)

and we get the Laplace (cofactor) expansion

$$\det D = \sum_j D_{ji} C_{ji} , \ \forall i .$$  \hspace{1cm} (188)

Move the $i$th particle: the $i$th row of $D$ changes, but $C_{ij}$ doesn’t, so

$$q_i = \frac{\det D^{new}}{\det D^{old}} = \frac{\sum_j D_{ji}^{new} C_{ji}^{old}}{\det D^{old}} = \frac{\sum_j D_{ji}^{new} (\det D^{old} \bar{D}_{ji}^{old})}{\det D^{old}}$$

$$= \sum_j \bar{D}_{ji}^{old} D_{ji}^{new} = \sum_j \bar{D}_{ji}^{old} \phi_j (r_{i}^{new}) .$$  \hspace{1cm} (189)
Dealing with Slater determinants

Using Laplace expansion again,

\[
\frac{\nabla_i \det D}{\det D} = \frac{\sum_j C_{ji} \nabla_i D_{ji}}{\det D} = \frac{\sum_j \det D \tilde{D}_{ji} \nabla_i D_{ji}}{\det D} = \sum_j \tilde{D}_{ji} \nabla_i D_{ji}, \quad (190)
\]

so

\[
\frac{\nabla_i \det D}{\det D} = \sum_j \tilde{D}_{ji} \nabla_i \phi_j(x_i). \quad (191)
\]
Easy start with basis sets
The Basis Set Exchange (link to portal or link to github):^25

```plaintext
# STO-3G EMSL Basis Set Exchange Library 11/12/18 10:13 PM
# Elements References
# -------- ----------
# Na - Ar: W.J. Hehre, R. Ditchfield, R.F. Stewart, J.A. Pople,
# K,Ca - : W.J. Pietro, B.A. Levy, W.J. Hehre and R.F. Stewart,
#
$basis*
*
li STO-3G
*

3 s
  16.1195750 0.15432897
  2.9362007 0.53532814
  0.7946505 0.44463454

3 s
  0.6362897 -0.09996723
  0.1478601 0.39951283
  0.0480887 0.70011547

3 p
  0.6362897 0.15591627
  0.1478601 0.60768372
  0.0480887 0.39195739
*
$end
```

^25 Turbomole format.
Easy start with basis sets

The STO-3G basis set gives *three primitive Gaussians* (hence 3G) fit to a Slater-type orbital (hence STO); the s-functions are (always atomic units)

\[
\begin{align*}
\chi_1(r) &= 0.15432897 \times e^{-16.1195750r^2} \\
\chi_2(r) &= 0.53532814 \times e^{-2.9362007r^2} \\
\chi_3(r) &= 0.44463454 \times e^{-0.7946505r^2} .
\end{align*}
\] (192)

From the primitives we get *contracted Gaussians*,

\[
G_n(r) = \sum_{\alpha=1}^{N_n} \chi_\alpha(r) ,
\] (193)

and from contracted Gaussians the orbitals (need normalization),

\[
\phi_i(r) = \sum_{n=1}^{N_{basis}} c_{in} G_n(r) .
\] (194)

DMC and chemical accuracy

"Chemical accuracy" 1 kcal/mol

1 Hartree = 627.509 kcal/mol
1 kcal/mol = 0.00159360 Hartree

Where are we now:26

FIG. 4. Deviation of the DMC atomization energies from experiment for a single determinant SJ trial wavefunction composed of VMC optimized orbitals and a CAS SJ trial wavefunction. The MAD from experiment for the single determinant SJ trial wavefunction is 2.1 kcal/mol. The MAD from experiment for the CAS SJ trial wavefunction is 1.2 kcal/mol.

Let’s turn our attention to the statistical physics of equilibrium quantum systems. In canonical ensemble\(^{27}\) everything can be computed from the canonical partition function

\[
Z(\beta) = \sum_i e^{-\beta E_i} = \sum_i \langle i | e^{-\beta \hat{H}} | i \rangle \quad (195)
\]

where \(\hat{H} | i \rangle = E_i | i \rangle\) and \(\beta = 1/(k_B T)\), \(k_B\) is the Boltzmann constant, and \(T\) is the temperature. Now \(Z\) is the trace (norm) of the density operator

\[
\hat{\rho} \equiv e^{-\beta \hat{H}}, \quad \rho(R', R; \beta) \equiv \langle R' | e^{-\beta \hat{H}} | R \rangle, \quad (196)
\]

where the latter is the density matrix. Trace is basis-independent: Insert position eigenstates,

\[
\begin{align*}
Z(\beta) &= \int dR \sum_i \langle i | e^{-\beta \hat{H}} | R \rangle \langle R | i \rangle = \int dR \sum_i \langle R | i \rangle \langle i | e^{-\beta \hat{H}} | R \rangle \\
&= \int dR \langle R | e^{-\beta \hat{H}} | R \rangle = \int dR \rho(R, R; \beta). \quad (197)
\end{align*}
\]

\(^{27}\)Energy and particle number are conserved.
Path integral Monte Carlo (PIMC)

In most cases we don’t know $\langle R | e^{-\beta \hat{H}} | R \rangle$ – except for small $\beta$.\textsuperscript{28} Split the operator ($\hat{H}$ obviously commutes with $\hat{H}$, so this splitting is exact),

$$Z(\beta) = \int dR \langle R | e^{-\beta \hat{H}} | R \rangle = \int dR \langle R | e^{-\frac{\beta}{2} \hat{H}} e^{-\frac{\beta}{2} \hat{H}} | R \rangle .$$ \hspace{1cm} (198)

Repeat and insert position states,

$$Z(\beta) = \int dR dR_1 \ldots dR_{M-1} \langle R | e^{-\frac{\beta}{M} \hat{H}} | R_1 \rangle \langle R_1 | e^{-\frac{\beta}{M} \hat{H}} | R_2 \rangle \ldots \langle R_{M-1} | e^{-\frac{\beta}{M} \hat{H}} | R \rangle$$ \hspace{1cm} (199)

This is a multidimensional integral of a product of high-temperature density matrices

$$\rho(R', R; \frac{\beta}{M}) = \langle R' | \hat{\rho}(\frac{\beta}{M}) | R \rangle = \langle R' | e^{-\frac{\beta}{M} \hat{H}} \hat{\rho} | R \rangle .$$ \hspace{1cm} (200)

\textsuperscript{28}Small $\beta$ (high $T$) in PIMC is equivalent to the small timestep $\tau$ in DMC.
Primitive approximation

The high-temperature density matrix can be approximated using ideas described in the DMC section. The splitting \( e^{-\beta \hat{H}} \approx e^{-\beta \hat{T}} e^{-\beta \hat{V}} \) gives the primitive approximation for the high-\(T\) density matrix,

\[
\langle R' | e^{-\beta \hat{H}} | R \rangle \approx (4\pi \lambda / \beta)^{-3N/2} e^{-\frac{(R-R')^2}{4\lambda \beta}} e^{-\beta V(R)} .
\] (201)

In PIMC it’s customary to write the Hamiltonian as

\[
\hat{H} = -\lambda \nabla^2_R + V(R)
\] (202)

\[
\lambda \equiv \frac{\hbar^2}{2m} .
\] (203)

In DMC it’s also common to mark \( D \equiv \frac{\hbar^2}{2m} \), \( D \) for ”diffusion constant”.

With \( M \) time slices it’s nice to define

\[
\tau \equiv \frac{\beta}{M} .
\] (204)
Quantum-classical analogy

The partition function is an integral over all possible paths, which make a ring in imaginary time: \( R - R_1 - R_2 \ldots - R_{M-1} - R \). Define a link action

\[
S_m \equiv -\ln[\rho(R_m, R_{m+1}; \tau)] ,
\]

the exact density matrix is

\[
\rho(R_0, R_M; \beta) = \int dR_1 \ldots R_{M-1} \exp \left[ - \sum_{m=1,M} S_m \right].
\]

Customarily one defines the ”kinetic action”,

\[
K_m = \frac{3M}{2} \ln(4\pi \lambda \tau) + \frac{(R_{m-1} - R_m)^2}{4\lambda \tau}
\]

and the “inter-action” \( U_m = S_m - K_m \) (often symmetrized), so that the primitive action is

\[
S^{(\text{prim.})}_m = \frac{3M}{2} \ln(4\pi \lambda \tau) + \frac{(R_{m-1} - R_m)^2}{4\lambda \tau} + \frac{\tau}{2} \left[ V(R_{m-1}) + V(R_m) \right].
\]

“Spring term”

“Potential term”
Ring polymers

In polymer analogy the spring term holds the polymer together and the potential term gives the inter-polymer interaction.

Example: 5 particles and 4 time slices (not in scale)

Note: The springs in the figure represent the *kinetic*, spring action \( \frac{(R_{m-1} - R_m)^2}{4\lambda\tau} \), not the potential!
Ring polymers

Another view of 5 particles in 4 time slices:

Only particles on the same time slice interact:
Blues interact and reds interact, but blues don’t interact with reds.
Bosons and fermions

We haven’t touched the subject of particle statistics yet. The Hamiltonian (202) doesn’t know about statistics, we have to force it. The density matrix we used earlier was for *boltzmannons*\(^{29}\)

\[
\rho(R', R; \tau) = \langle R' | e^{-\beta \hat{H}} | R \rangle = \sum_i e^{-\beta E_i} \phi_i(R') \phi_i(R),
\]

so all we need is to rewrite this so that the \(\phi_i(R)\) are (anti)symmetric. We could do this with Slater determinants (fermion) or permanents (boson), but the notation is simpler with permutations,

\[
\phi_{i,B/F}(R) = \frac{1}{N!} \sum_P (\pm 1)^P \phi_i(\hat{P}R) \quad \begin{cases} + & \text{for bosons (B)} \\ - & \text{for fermions (F)} \end{cases},
\]

\(\hat{P}\) permutes particle labels and \(P\) counts pair permutations, and \(\sum_P\) denotes the sum over all permutations. For \(N = 3\):

\[
\begin{align*}
123 : & \quad P = 0 \\
213 : & \quad P = 1 \\
231 : & \quad P = 2 \\
312 : & \quad P = 1 \\
312 : & \quad P = 2
\end{align*}
\]

\(^{29}\)Boltzmannons are distinguishable particles.
Bosons and fermions

Working backwards, we get the Bose/Fermi density matrix

\[
\rho_{B/F}(R', R; \tau) = \sum_i e^{-\beta E_i} \phi_i(R') \left[ \frac{1}{N!} \sum_P (\pm 1)^P \phi_i(\hat{P} R) \right]
\] (211)

The symmetrized states give the boson density matrix,

\[
\rho_B(R', R; \tau) = \frac{1}{N!} \sum_P \rho(R', \hat{P} R; \tau)
\] (212)

and the antisymmetrized states give the fermion density matrix,

\[
\rho_F(R', R; \tau) = \frac{1}{N!} \sum_P (-1)^P \rho(R', \hat{P} R; \tau)
\] (213)

It doesn't matter whether the permutation operator operates on \(R\), on \(R'\), or on both.
Let's go back on the ring polymer picture. How large are the rings? The spring term is a gaussian $e^{\frac{(R_{m-1}-R_m)^2}{4\lambda\tau}}$ with variance $\sigma^2 = 2\lambda\tau$, and $M$ of them could cover a circle with radius

$$r \sim \sqrt{2\lambda M\tau} = \sqrt{\frac{\hbar^2\beta}{m}}. \quad (214)$$

Multiply with $\sqrt{2\pi}$ and you get the deBroglie wavelength,

$$\lambda_{\text{deBroglie}} = \sqrt{\frac{2\pi\hbar^2\beta}{m}}. \quad (215)$$

Lower $T \Rightarrow$ larger $\beta \Rightarrow$ larger $\lambda_{\text{deBroglie}}$: at some low $T$ two boson polymers may coil together. The inter-polymer spacing in 3D is about $\rho^{-1/3}$, so quantum statistics is important below the degeneracy temperature

$$T_D = \frac{\rho^{2/3}\hbar^2}{mk_B} \quad (3D). \quad (216)$$

Remember the periodic boundaries in space.
Winding number and superfluidity

- Experimental superfluidity (1938): Kapitza, Allen and Misener
- Superfluid density is directly related to the response of the free energy to a boundary phase twist. \(^{31}\)
- PIMC on liquid \(^4\)He: Ceperley and Pollock.\(^{32}\)
- Lots of misconceptions and assumptions are about. \(^{33}\) The winding number is undefined for Hamiltonians that don’t conserve the particle number.

**Experimental setup:** System between two cylinders of radii \(R\) and \(R + d\), and rotate the cylinders

**PIMC setup:** assume \(d \ll R\) and put the system between two walls that are moving with velocity \(v\) wrt. to the liquid.

**Two-fluid model:** The *normal component*, with density \(\rho_N\), responds to the motion of the walls: friction. The non-responding part is *superfluid* \(\rho_s\).


Winding number and superfluidity

A gas of excitations with energy $\epsilon(p)$ carries momentum (Galilei transf.)

$$\langle \hat{P} \rangle_v = \int dr \ p n(\epsilon - p \cdot v) ,$$

(217)

where $n(\epsilon)$ is the distribution function for the gas at rest.$^{34}$ For small $v$ you can expand and get $\langle \hat{P} \rangle_v = \text{something} \times v$, the something is the mass associated with excitations - this is the mass of the normal fluid.

Follow Pollock ans Ceperley and compute the change in free energy. In the rest frame walls moving with velocity $v$

$$\hat{\mathcal{H'}} = \sum_j \frac{(-i\hbar \nabla_j - mv)^2}{2m} + \hat{\mathcal{V}} ,$$

(218)

and the density operator is $\hat{\rho'} = e^{-\beta \hat{\mathcal{H'}}}$, the same as $\hat{\rho}_v$ in the rest frame of the system. The momentum of the system can now be computed.

Winding number and superfluidity

Using statistical physics,

\[
\langle \hat{A} \rangle = \frac{\text{Tr}[\hat{A}\hat{\rho}]/\text{Tr}[\hat{\rho}]}{\sum_i \langle i | \hat{\rho} | i \rangle} = \frac{1}{Z} \sum_i \langle i | \hat{\rho} | i \rangle,
\]

(219)

after some work we find the momentum of the system with moving walls (total density \(\rho\), volume is \(N/\rho\)),

\[
\langle \hat{P} \rangle_v = \frac{\text{Tr}[\hat{P}\hat{\rho}_v]/\text{Tr}[\hat{\rho}_v]}{\text{normal mass } \times v = (N_N m)v = \left(\frac{N}{\rho} \frac{N}{\rho} m\right)v}.
\]

(220)

Apparently this proportionality can hold only if \(v\) is small, but that’s all we need.
Winding number and superfluidity

Compute the momentum and expand it to linear order in \( \nu \). The free energy is

\[
F_\nu = -\frac{1}{\beta} \ln Z_\nu = -\frac{1}{\beta} \ln \text{Tr}[\hat{\rho}_\nu] ,
\]

so you finally get (details in Pollock-Ceperley article),

\[
\Delta F_\nu \frac{N}{\rho} = \frac{1}{2} mv^2 \frac{\rho S}{\rho} + \mathcal{O}(\nu^4) .
\]

In PIMC we don’t need actual moving walls to measure \( \Delta F_\nu \). Notice that the density matrix with moving walls, \( \rho_\nu \), satisfies the Bloch equation,

\[
- \frac{\partial \rho_\nu(R', R; \beta)}{\partial \beta} = \hat{H}' \rho_\nu(R', R; \beta) .
\]

where we use periodic boundary conditions, \( r_i + L = r_i \).
Winding number and superfluidity

Define a modified density matrix $\tilde{\rho}$ via a phase factor,

$$\phi(R) \equiv \frac{m}{\hbar} \sum_j (r_j - r'_j)$$

$$\rho_v(R', R; \beta) = e^{i\phi(R)} \tilde{\rho}(R', R; \beta),$$

and insert it to the Bloch equation. After some manipulation you get

$$-\frac{\partial \tilde{\rho}(R', R; \beta)}{\partial \beta} = \hat{H} \tilde{\rho}(R', R; \beta),$$

but the boundary conditions come with a phase factor,

$$\tilde{\rho}(R', r_1...r_i + L, ...r_N; \beta) = e^{-imv \cdot L / \hbar} \tilde{\rho}(R', R; \beta).$$

The **winding number** $W$ counts how many times particle paths wind around periodically. Each winding adds a factor $e^{-imv \cdot L / \hbar}$. 

\[\text{Quantum Monte Carlo (QMC)}\]

November 14, 2018 114 / 195
Winding number and superfluidity

BTW, the principle how the Galilei transformation

\[-i\hbar \nabla j - mv\]  \hspace{1cm} (228)

in the Hamiltonian turns into to a boundary condition for the density matrix, phase \(e^{-imv \cdot L/\hbar}\), is widely applicable. Electrons in a magnetic field have the Hamiltonian

\[-i\hbar \nabla j - eA,\]  \hspace{1cm} (229)

and if they move a full circle round a flux tube with flux \(\Phi\), their wave function picks up the *Aharonov-Bohm phase* \(e^{-ie\Phi/\hbar}\). \footnote{The funny thing is that the circulating electron may never be anywhere where \(A \neq 0\), and still it’s wave function picks up the phase.}
Winding number and superfluidity

From the winding number we can compute the superfluid fraction (in $d$ dimensions),

\[
\frac{\rho_S}{\rho} = \frac{m \langle W^2 \rangle L^{2-d}}{\hbar^2 \rho d \beta}.
\]  

(230)

The free Bose gas superfluid transition temperature is

\[
T_c = \frac{3.31\hbar^2}{gmk_B} \rho^{2/3},
\]  

(231)

where $g$ is the $k$-state degeneracy factor. For liquid $^4$He this would give $T_c \sim 3.14$ K, \(^{36}\) but He-He interactions suppress it to 2.17 K. Let’s see visually what superfluidity requires, next pages contain imaginary time plots of paths.

---

Boson superfluidity: not superfluid
Boson superfluidity: not superfluid
Boson superfluidity: not superfluid
Boson superfluidity: not superfluid
Boson superfluidity: not superfluid
Boson superfluidity: Start again, not superfluid
Boson superfluidity: Superfluid!
Boson superfluidity: \textit{Superfluid!}

- Net flux in imaginary time over \textit{any} boundary in space (such as the green boundary, periodic boundaries are not special)
- Four particles in exchange loop, one particle not participating
- Order $N$ particles needed for the exchange loop to reach through the box, but not all of them.
Pollock and Ceperley (1987):

FIG. 1. Computed $\rho_N/\rho$ values along the SVP curve (lower panel). The solid line is the experimental curve (Ref. 13). The solid dots, for 125 atoms, are calculated from Eq. (33). The estimates based on the winding number [Eq. (22)] (open circles) are for 64 atoms. The upper panel shows the $x$-direction winding-number distribution at $T=2.0$ and 1.18 K.
Liquid $^4$He

D. Ceperley, Rev. Mod. Phys. 67, 279 (1995):

**FIG. 10.** The extended trace of six $^4$He atoms at a temperature of 0.75 K and with 53 time slices. The dashed square represents the periodic boundary conditions. Three of the atoms are involved in an exchange which winds around the boundary in the $x$ direction.

**FIG. 11.** The specific heat of $^4$He: solid line, experiment at saturated vapor pressure (Wilks, 1967); triangles with error bars, PIMC calculations (Ceperley and Pollock, 1986); open circles, Feynman-Kikuchi model with $20^3$ sites (Elser, 1984). In Elser’s calculation, only the fluctuation term in the specific heat has been included, and the temperature has been scaled to match the experimental transition temperature.
Liquid $^4$He

D. Ceperley, Rev. Mod. Phys. 67, 279 (1995):

![Graph showing energy per atom of $^4$He vs temperature.](image)

**FIG. 13.** The energy/atom of $^4$He: solid line, experiment at saturated vapor pressure; symbols with error bars, PIMC calculations (Ceperley and Pollock, 1986). The upper panel is the potential energy (solid line and left scale) and kinetic energy (dashed line and right scale).
Liquid $^4$He

The density has the kink, so it’s present ”already” in the liquid structure:

\[ \frac{\langle V \rangle}{N} = N^{-1} \int d\mathbf{R} |\psi(\mathbf{R}; T)|^2 \sum_{i<j=1}^{N} V(r_{ij}) / \int d\mathbf{R} |\psi(\mathbf{R}; T)|^2 \]

\[ = N^{-1} \int dr_1 dr_2 V(r_{12}) \frac{N(N-1)}{2} \int dr_3 \ldots dr_N |\psi(\mathbf{R}; T)|^2 / \int d\mathbf{R} |\psi(\mathbf{R}; T)|^2 \]

\[ = N^{-1} \Omega(T) 4\pi \rho(T)^2 \int_0^\infty dr r^2 V(r) g(r; T) = 4\pi \rho(T) \int_0^\infty dr r^2 V(r) g(r; T) \]  \hspace{1cm} (232)

The potential energy per particle is

so comparing with the PIMC data, most of the $T$-dependence comes from $\rho(T)$, the rest is roughly a constant, $\sim -978 \text{ KÅ}^{-3}$. The pair distribution $g(r, T)$ has a very weak temperature dependence.

Path sampling

We want to sample different terms in $Z$ as effectively as possible, including permutations of indistinguishable particles.

- Single bead move: move just one random particle in a random time slice
- Rigid move: move the whole imaginary time world line of a random particle
- Recreate path(s) for multiple time slices, say slices 2, 3, 4, keeping slices 1 and 5 fixed accept or reject at any stage:
  - Bisection: sample 3 as midpoint 1-3-5, sample 2 as midpoint 1-2-3 and 4 as midpoint 3-4-5.
    - Easy permutation sampling, but insufficient for some topologies
  - Staging: sample 2 from 1-2-5, then 3 from 2-3-5, then 4 from 3-4-5.
- Worm move: open a path and let the ends move and recombine
  - Global updates of paths
  - *The ultimate permutation sampler*
Path sampling: single-bead move

Update: two $G^*$'s and potential energy in the time slice. *Ineffective but cheap*
Path sampling: rigid move

Update: potential energy of all time slices. No $G\hat{\tau}$ change.

Effective but expensive
Path sampling: bisection
Path sampling: bisection
Path sampling: bisection
Path sampling: bisection
Path sampling: bisection
Path sampling: bisection
Path sampling: Adding permutations to bisection
Path sampling: Adding permutations to bisection
Path sampling: Adding permutations to bisection
Path sampling: Adding permutations to bisection
Path sampling: Adding permutations to bisection
Path sampling: Adding permutations to bisection
Path sampling: Adding permutations to bisection
Bisection

Comments on the bisection update

- Effective, path changes a lot
  - If the new midpoint is accepted, it may be far from the original point
  - If the new midpoint is rejected (large potential) we don’t waste time on the rest
- Does the algorithm early-reject a move that could be finally accepted? In principle yes, but without any practical consequences

- Move to the temporary mid-point (red) is deterministic
- Move from the temporary mid-point to new point (green) can be done using random Gaussian move. The length can be adjusted for optimal acceptance.
- Long $G\hat{\tau}$’s of previous stage can be divided away in the next stage. Intermediate steps can be free-spirited, but in the final step you have to use the correct weights.
- Adjustable parameters: length of the re-sampled path and amplitudes of random Gaussian moves on each bisection level.

---

37 Llorenç Brualla i Barberà, Thesis (Barcelona, 2002).
Worm

The worm algorithm\(^{38}\) owns to the advances made in the Ising model MC sampling in the 80’s. It solves many problems in PIMC path sampling:

- **SOLVED: Global changes** are more efficient than local ones
- **SOLVED: Long permutation loops** are easily created with high acceptance
  - other path sampling methods *explicitly construct* permutations
  - acceptance of suggested permutations is very low \(\Rightarrow\) permutation sampling is very inefficient
  - on a torus-shaped space you practically never get a path winding around \(\Rightarrow\) wrong result
- **SOLVED: Evaluation of off-diagonal quantities**: condensate fraction and imaginary-time correlations, without any ambiguity in normalization.

Worm

Start with a configuration that describes one possible term in the partition function $Z$. This term is in the $Z$ sector. The basic worm updates are:

1. **Open**: Open a path, this leaves two dangling ends, *Ira* and *Masha*. You generated a term that’s not in $Z$, and entered the $G$ sector.

2. **Advance/recede**: Ira moves ahead (pulls back) and the worm gets longer (shorter)

3. **Insert/remove**: Add a completely new piece of path or remove one

   $\Rightarrow$  *Grand Canonical Ensemble* (variable $N$, fixed chemical potential $\mu$)

4. **Swap**: Ira jumps to another path, close the old path

5. **Close**: Close the dangling ends of the worm and return to the $Z$ sector. This must be in balance with the open update – *detailed balance!*

While in the $Z$ sector, measure anything you would from $Z$.

While in the $G$ sector, measure off-diagonal quantities: Ira and masha correspond to the field operators $\psi(r; \tau)$ and $\psi^\dagger(r'; \tau')$. 
Path sampling: worm open

\[ \hat{\psi}^+ (\mathbf{r}; \tau_M) \quad \hat{\psi} (\mathbf{r}; \tau_I) \]
Path sampling: worm recede
Path sampling: worm advance
Path sampling: worm close
Path sampling: worm insert
Path sampling: worm advance
Path sampling: worm advance
Path sampling: worm advance
Path sampling: worm advance
Path sampling: worm swap
Path sampling: worm swap *PERMUTATIONS!*

![Diagram of worm swap](image-url)
Worm partition function

The $Z$ and $G$ sectors are also known with less colourful names: *diagonal* and *off-diagonal* sector. Ira and Masha are known as *head and tail*. The worm samples an ”extended” partition function, ($I, M$ are Ira and Masha)

$$Z_{\text{worm}} = \sum_{N=0}^{\infty} Z_N \left[ e^{\beta \mu N} + C \sum_{i_I, i_M} \int d r_I d r_M G(r_I, r_M; (i_I - i_M)\tau) \right],$$

(233)

where the integrand is the *Matsubara Green’s function* (thermal average),

$$G(r, r'; \tau - \tau') = \langle \hat{T}_\tau \hat{\psi}(r, \tau) \hat{\psi}^\dagger(r'; \tau') \rangle.$$  

(234)

Here $\hat{T}_\tau$ is the imaginary-time ordering operator and $Z_N$ is the canonical partition function. The $Z_{\text{worm}}$ is sum of the grand canonical partition function and sum over all worms.
Worm partition function

- Worm’s natural ensemble is grand canonical, but also a canonical implementation has been done. The grand canonical partition function is

$$Z_G = \text{Tr} \left[ e^{-\beta (\hat{H} - \mu N)} \right] = \sum_{N=0}^{\infty} Z_N e^{\beta u N},$$

and we approximate the canonical partition function $Z_N$ it with the multidimensional integral, $(X \equiv \{R_1, R_2...R_M\})$

$$Z_G \approx \sum_{N=0}^{\infty} e^{\beta u N} \int dX A(X; \tau) e^{-U(X)} .$$

Here $A(X; \tau) = \rho_0(R_1, R_2; \tau) - \rho_0(R_{M-1}, R_M; \tau)$ is the free-particle contribution and $U(X)$ is the rest, all interaction terms.

- Paths (world lines) are created sampling exactly the free-particle density matrices $\rho_0(r_1, r_2; \tau)$. This can be done using staging.
Staging algorithm

We generate a random walk path between two known endpoints $r_1$ and $r_m$, marked just 1 and $m$. The imaginary-time difference is $\tau_m - \tau_1 \equiv \tau_{1m}$.

Split a long-leap Gaussian to short-step Gaussians:

$$\rho_0(r_1, r_m; \tau_{1m}) \equiv \langle r_1 | e^{-\tau_{1m} \hat{T}} | r_m \rangle = \frac{1}{(4\pi\lambda\tau_{1m})^{dN/2}} e^{-\frac{(r_1 - r_m)^2}{4\lambda\tau_{1m}}}$$ (237)

$$= \int dr_2 \ldots dr_{m-1} \langle r_1 | e^{-\tau_{12} \hat{T}} | r_2 \rangle \langle r_2 | e^{-\tau_{23} \hat{T}} | r_3 \rangle \ldots \langle r_{m-1} | e^{-\tau_{m-1,m} \hat{T}} | r_m \rangle$$

$$= \int dr_2 \ldots dr_{m-1} \frac{1}{(4\pi\lambda\tau_{12}\tau_{23}\ldots\tau_{m-1,m})^{dN/2}} e^{-\frac{(r_1 - r_2)^2}{4\lambda\tau_{12}}} e^{-\frac{(r_2 - r_3)^2}{4\lambda\tau_{23}}} \ldots e^{-\frac{(r_{m-1} - r_m)^2}{4\lambda\tau_{m-1,m}}} \mathcal{N}.$$  

The unknown, intermedia coordinates $r_2, \ldots, r_{m-1}$ are entangled, but we can re-arrange terms.
\[
\rho_0(r_1, r_m; \tau_{1m}) = \mathcal{N} \int dr_2 \ldots r_{m-1} e^{-\frac{(r_1 - r_m)^2}{4\lambda \tau_{1m}}}
\]

\[
\begin{bmatrix}
\frac{-(r_1-r_2)^2}{4\lambda \tau_{12}} & \frac{-(r_2-r_m)^2}{4\lambda \tau_{2m}} & \frac{-(r_3-r_m)^2}{4\lambda \tau_{3m}} & \frac{-(r_4-r_m)^2}{4\lambda \tau_{4m}} \\
\frac{-(r_1-r_m)^2}{4\lambda \tau_{1m}} & e & e & e \\
\frac{-(r_{m-3}-r_{m-2})^2}{4\lambda \tau_{m-3,m-2}} & \frac{-(r_{m-2}-r_m)^2}{4\lambda \tau_{m-2,m}} & e & e \\
e & e & e & e
\end{bmatrix}
\]

\[
\ldots \left[ \frac{-(r_{m-3}-r_{m-2})^2}{4\lambda \tau_{m-3,m-2}} \right] \left[ \frac{-(r_{m-2}-r_m)^2}{4\lambda \tau_{m-2,m}} \right] \left[ \frac{-(r_{m-1}-r_m)^2}{4\lambda \tau_{m-1,m}} \right]
\]

and all extra colored terms cancel. The brackets simplify, for example

\[
\begin{bmatrix}
\frac{-(r_1-r_2)^2}{4\lambda \tau_{12}} & \frac{-(r_2-r_m)^2}{4\lambda \tau_{2m}} \\
\frac{-(r_1-r_m)^2}{4\lambda \tau_{1m}} & e
\end{bmatrix}
= e^{\frac{-(r_2-r_2^*)^2}{4\lambda \tau_{12m}}}
\]

\[
\tau_{12m} \equiv \frac{\tau_{12} \tau_{2m}}{\tau_{1m}} \quad r_2^* \equiv \frac{\tau_2 m r_1 + \tau_{12} r_m}{\tau_{1m}}.
\]
Staging algorithm

We now have a formula to generate a random walk between fixed end points \( r_1 \) and \( r_m \) – don’t forget the blue term! –

\[
\rho_0(r_1, r_m; \tau_{1m}) = e^{-\frac{(r_1-r_m)^2}{4\lambda\tau_{1m}}} \mathcal{N} \int dr_2...dr_{m-1} e^{-\frac{(r_2-r_2^*)^2}{4\lambda\tau_{12m}}} ... e^{-\frac{(r_{m-1}-r_{m-1}^*)^2}{4\lambda\tau_{m-2,m-1,m}}},
\]

where

\[
\tau_{ijk} \equiv \frac{\tau_{ij}\tau_{jk}}{\tau_{ik}}, \quad \tau_{ij} \equiv \tau_{j} - \tau_{i} \quad (241)
\]

\[
r_j^* \equiv \frac{\tau_{jk}r_i + \tau_{ij}r_k}{\tau_{ik}}. \quad (242)
\]

Usage: compute \( r_2^* \) using \( r_1 \) and \( r_m \), and set

\[
r_2 = r_2^* + \sqrt{2\lambda\tau_{12m}}\eta. \quad (243)
\]

Compute \( r_3^* \), sample \( r_3 \), etc. This samples the integrand exactly.

Notice that this staging algorithm doesn’t assume that the imaginary times are evenly spaced.
Worm partition function

The acceptance probabilities of worm updates are given by the Metropolis-Hastings formula:

$$P_{old \rightarrow new} = \min \left\{ 1, \frac{T(new \rightarrow old)W(new)}{T(old \rightarrow new)W(old)} \right\}.$$  \hspace{1cm} (244)

Free-particle paths between fixed $r'$ and $r$ are created with a random walk, which samples the product

$$\rho_0(r', r_1; \tau)\rho_0(r_1, r_2; \tau)\ldots\rho_0(r_{k-1}, r; \tau).$$  \hspace{1cm} (245)

From the $Z_{worm}$ in Eq. (233) we should be able to compute the update probabilities.

Let’s go through the worm updates one by one. For more details, see the article by Boninsegni et al. from 2006.
Open worm

Adjustable parameters are marked with blue. From now on, above means up, and below means down in imaginary time: imaginary time is periodic, with period $\beta$.

Open

1. Set Ira to a random bead, there are $N_{\text{beads}}$ to choose from.
2. Pick $k \in U[1, K]$ and set Masha $k$ steps above Ira; $\tau_{IM} \equiv \tau_M - \tau_I$.
3. Remove beads between Ira and Masha.
4. Accept with probability

$$P_{\text{open}} = \min \left\{ 1, \frac{CKN_{\text{beads}}e^{-\Delta U - \mu \tau_{IM}}}{\rho_0(r_I, r_M; \tau_{IM})} \right\}.$$  \hspace{1cm} (247)

where

$$\Delta U = U(\text{new}) - U(\text{old}) = U(\text{open}) - U(\text{close}).$$
Worm open, probability details

Let’s look how this creates the worm term in Eq. (233), repeated here in a
simplified form,

\[
Z_{\text{worm}} = \sum_{N=0}^{\infty} \left[ Z_N e^{\beta \mu N} + C \sum_{i_l, i_M} \int dr_I dr_M (Z_N e^{\beta \mu N} - \text{path I-M}) \right].
\]

Ira was picked from \( N_{\text{beads}} \) beads, which is the current value of \( NM \); and
the reverse can be done in 1 way. Masha was picked from \( K \) beads,
reverse has 1 possible way. Putting together, we have so far

\[
\frac{T(\text{new} \rightarrow \text{old})}{T(\text{old} \rightarrow \text{new})} = KN_{\text{bead}} = KNM.
\]

(248)

The old weight is \( e^{\beta \mu N} \), while the new weight has a factor \( C \) and there are
\( N - 1 \) particles for the duration \( \tau_{IM} \),
\( e^{\mu N(\beta - \tau_{IM}) + \mu (N-1) \tau_{IM}} \), so

\[
\frac{W(\text{new})}{W(\text{old})} = Ce^{-\mu \tau_{IM}}.
\]

(249)
Open worm, details

Now things gets interesting. Deleting the path between Ira and Masha also deletes the interactions this particle had between Ira and Masha. Those are present in $Z_N$, so the weights have changed by the amount of the potential action; the explicit form depends on the approximation. Using $U$ defined in Eq. (236), and marking $\Delta U \equiv U(\text{new}) - U(\text{old})$, we get

$$\frac{W(\text{new})}{W(\text{old})} = e^{-\Delta U}.$$  \hspace{1cm} (250)

The free-particle paths are also different,

$$\ldots \rho_0(r_{I-1}, r_I; \tau) \rho_0(r_I, r_{I+1}; \tau) \ldots \rho_0(r_{M-1}, r_M; \tau) \rho_0(r_M, r_{M+1}; \tau) \ldots .$$

\underline{cut off in open update}

In closing the path we recreate the path segment with staging, which has the Ira—Masha factor that’s not taken care of:

$$\frac{W(\text{new})}{W(\text{old})} = \frac{1}{\rho_0(r_I, r_M; \tau_{IM})}.$$ \hspace{1cm} (251)

Now we have covered all terms in Eq. (247).
Open worm, details

- If $C = 0$ we never open a worm. Fine, because then there is none.
- If $C$ increases, we accept more worms.
- More particles, more worms – unless $C$ cancels $N$.
- More time slices, more worms – unless $C$ cancels $M$.
- We accept less worms with Ira and Masha farther apart in a random walk.
Close worm

Reverse of open, must satisfy the detailed balance condition.

Close

1. Reject the update, if Masha is zero or more than $K$ steps above Ira.
2. Generate a path from Ira to Masha.
3. Accept with probability

$$P_{\text{close}} = \min \left\{ 1, \frac{\rho_0(r_I, r_M; \tau_{IM})}{CKN_{\text{bead}} e^{\Delta U - \mu \tau_{IM}}} \right\},$$ (253)

where $N_{\text{bead}}$ is the number of beads after closing, and

$$\Delta U = U(\text{new}) - U(\text{old}) = U(\text{closed}) - U(\text{open}),$$

hence it’s $-\Delta U$ in used in the opening acceptance.
The probability of closing a worm may get very small if the random walk distance of Ira and Masha is large, i.e., $\rho_0(r_I, r_M; \tau_{IM})$ is very small. Such worms are rarely created, but they are also hard to close. In order to avoid these sticky worms Boninsegni et al. used a hard limit, that keeps $\rho_0(r_I, r_M; \tau_{IM})$ (actually $\ln[\rho_0(r_I, r_M; \tau_{IM})]$) more reasonable: If

$$\frac{-(r_I - r_M)^2}{4\lambda \tau_{IM}} > X$$

opening and closing are rejected.$^{39}$ This choice is deterministic, but also symmetric, so $T(new \rightarrow old) = T(old \rightarrow new)$ and no harm is done to stochasticity.

$^{39}$Boninsegni et al. used $X = 4$ in liquid He calculations.
This seeds a new world line strand of length $k$.

**Insert**

1. Set Masha to a random $r$ from the simulation volume $\Omega$ and random time slice.
2. Pick $k \in U[1, K]$ generate a $k$-step path above Masha; Put Ira there.
3. Accept with probability

$$P_{\text{insert}} = \min \left\{ 1, CKM\Omega e^{\Delta U - \mu_{IM}} \right\}, \quad (256)$$
Remove worm

The reverse update of insert.

Remove

1. If the worm length is $> K$ reject remove, the worm is too long.
2. Accept with probability

$$P_{\text{remove}} = \min \left\{ 1, \frac{e^{\Delta U - \mu T_{IM}}}{CKM\Omega} \right\} ,$$  \hfill (258)
This moves Ira up in imaginary time.

**Advance**

1. Pick \( k \in U[1, K] \)
2. Generate a \( k \)-step path from old Ira up in imaginary time and set the new Ira there.
3. Accept advance with probability

\[
P_{\text{advance}} = \min \left\{ 1, e^{\Delta U + \mu \tau IM} \right\}.
\]  

(260)
This moves Ira down in imaginary time.

**Recede**

1. Pick \( k \in U[1, K] \). If the worm has less than \( k \) beads reject the update.
2. Move Ira \( k \) steps down in imaginary time, delete beads on the way.
3. Accept with probability

\[
P_{\text{recede}} = \min \left\{ 1, e^{\Delta U - \mu \tau_{IM}} \right\}, 
\]

(262)
1. Pick \( k \in U[1, K] \). Build a list \( \{S_i\} \) of beads \( k \) steps above Ira on another world line.

2. Build a weight list \( \{\rho_0(\mathbf{r}_I, \mathbf{r}_{S_i}, \tau_{IS_i})\} \). Normalization of the list is \( \sum_S = \sum_i \rho_0(\mathbf{r}_I, \mathbf{r}_{S_i}, \tau_{IS_i}) \).

3. Pick a bead \( S \) with a probability of the normalized weight list.

4. Find the bead \( T \) that is \( k \) steps down in imaginary time from bead \( S \). If Masha is met along the way, reject the update.

5. Build another weight list \( \{\rho_0(\mathbf{r}_S, \mathbf{r}_{T_i}, \tau_{IT_i})\} \). Normalization of the list is \( \sum_T = \sum_i \rho_0(\mathbf{r}_S, \mathbf{r}_{T_i}, \tau_{ST_i}) \).

6. Generate a free-particle random path from Ira to \( S \).

7. Accept swap and move Ira to \( T \) with probability

\[
P_{\text{swap}} = \min \left\{ 1, \frac{\sum_S e^{\Delta U}}{\sum_T} \right\}. \tag{264}
\]
Swap details

The first weight list and sampling bead $S$ from it favors swap attempts to beads that are not too far in a random walk from Ira. This is $T(old \rightarrow new)$. Another weight list is needed because the swap is its own reverse, so we must compute also $T(new \rightarrow old)$. 
Some PIMC researchers

Just to give you a few pointers to some useful names. In no particular order:

► D. M. Ceperley, E. L. Pollock, B. Militzer, H. Kleinert
► V. S. Filinov, M. Bonitz, V. E. Fortov, T. Schoof, S. Groth
► J. Boronat, J. Casulleras, S. Giorgini
► M. Boninsegni, N. Prokof’ev, B. Svistunov, S. A. Chin, W. Janke, T. Sauer
► D. E. Galli, C.H. Mak, S. Zakharov, M. F. Herman, E. J. Bruskin, B. J. Berne
► Finland (TUT): T. T. Rantala and his students (I. Kylänpää, M. Leino, J. Tiihonen etc.)

And, of course, R. Feynman.
Survey of PIMC applications

- Uniform electron gas: very popular, see e.g. Filinov et al., Phys. Rev. E 91, 033108 (2015)

Excluding helium systems, almost everybody did/does them.
Fermion sign problem: the Nemesis of QMC.

The expectation value of some quantity $\hat{A}$ is

$$\langle \hat{A} \rangle = \frac{\text{Tr}[\hat{A} \hat{\rho}]}{\text{Tr}[\hat{\rho}]},$$

which is evaluated in MC using weights $w_i$,

$$\langle A \rangle_w = \frac{\sum_i w_i A_i}{\sum w_i},$$

but the weights $w_i$ may be negative, unsuitable for MC. Separate the sign as $w_i = \text{sgn}(w_i)|w_i|,$

$$\langle A \rangle_w = \frac{\sum_i w_i A_i}{\sum w_i} = \frac{\sum_i \text{sgn}(w_i)|w_i| A_i}{\sum \text{sgn}(w_i)|w_i|} = \frac{\sum_i |w_i| \text{sgn}(w_i) A_i}{\sum_i |w_i| \text{sgn}(w_i)}.$$

The factor

$$p_i \equiv \frac{|w_i|}{\sum_i |w_i|},$$

is a good probability, so we should just evaluate two averages. Easy, ha?
Fermion sign problem: the Nemesis of QMC.

Using the probability made of absolute values we get

\[
\langle A \rangle_w = \frac{\sum_i p_i \text{sgn}(w_i) A_i}{\sum_i p_i \text{sgn}(w_i)} = \frac{\langle \text{sign} A \rangle_{|w|}}{\langle \text{sign} \rangle_{|w|}}.
\] (269)

In principle, this can be easily done in MC, but there's a catch. The denominator is related to the partition function \(Z\) of weights \(w_i\), and another one of weights \(|w_i|\),

\[
\langle \text{sign} \rangle_{|w|} = \sum_i p_i \text{sgn}(w_i) = \frac{\sum_i w_i}{\sum_i |w_i|} = \frac{Z_w}{Z_{|w|}} = \frac{e^{-\beta F_w}}{e^{-\beta F_{|w|}}} = e^{-\beta(F_w - F_{|w|})},
\] (270)

where the free energy is \(F = -\frac{1}{\beta} \ln Z\) (thermodynamics). As always in MC, the error of the mean diminishes as \(1/\sqrt{M}\) with \(M\) samples, but there is a factor that grows,

\[
\Delta(\langle \text{sign} \rangle_{|w|}) = \frac{1}{\langle \text{sign} \rangle_{|w|}} \sqrt{\langle \text{sign}^2 \rangle_{|w|} - \langle \text{sign} \rangle_{|w|}^2} \sim \frac{e^{\beta(F_w - F_{|w|})}}{\sqrt{M}} = \frac{e^{\beta(N(f_w - f_{|w|})}}{\sqrt{M}}.
\] (271)
Fermion sign problem: the Nemesis of QMC.

The error behaves like

\[
\Delta\left(\langle \text{sign} \rangle_{|w|}\right) \sim \frac{e^{\beta N(f_w - f_{|w|})}}{\sqrt{M}},
\]

where \( f \)'s are free energies per particle – remember \( F \) is extensive. The difference \( f_w - f_{|w|} \) is positive, because the boson energy with weights \( |w| \) is below the fermion energy with weights \( w \). The factor grows exponentially both with decreasing temperature, and with increasing number of fermions \( N \). This is the **fermion sign problem**.

What of the nodes:

- **FN-DMC**: use fixed nodes that are good at \( T = 0 \).
- **PIMC**: \( T = 0 \) nodes and finite- \( T \) nodes are not the same. Instead, the density matrix nodes depend on the imaginary time:

\[
\rho(R', R_0; t) = 0 \text{ with fixed (reference point) } R_0 \text{ and } t.
\]
Fermion sign problem: strategies

- **Restricted PIMC (RPIMC):** Force nodes via a *trial density matrix* and keep paths from crossing to another sign
  - determinant of free-particle density matrices – accurate at high $T$
  - variational trial density matrix (Militzer and Ceperley)
  - Jellium: RPIMC does not work at high density ($r_s < 1$)  

- **Direct PIMC (DPIMC)**

- **Configuration PIMC (CPIMC)****
  - no timestep error: **continuous time** QMC
  - excellent at high density

- **Permutation Blocking PIMC (PB-PIMC)**

---

Fermion sign problem; perspectives

Troyer and Wiese \(^{44}\)

Quantum Monte Carlo simulations, while being efficient for bosons, suffer from the "negative sign problem" when applied to fermions—causing an exponential increase of the computing time with the number of particles. A polynomial time solution to the sign problem is highly desired since it would provide an unbiased and numerically exact method to simulate correlated quantum systems. Here we show that such a solution is almost certainly unattainable by proving that the sign problem is nondeterministic polynomial (NP) hard, implying that a generic solution of the sign problem would also solve all problems in the complexity class NP in polynomial time.

Fermion sign problem; perspectives

Troyer and Wiese study the NP hard problem of determining whether there exists a state with energy $E \leq E_0$, for a fixed bound $E_0$, of the Ising Hamilton function

$$H = - \sum_{\langle j,k \rangle} J_{jk} \sigma_j \sigma_k . \quad (274)$$

What makes this NP hard is the evaluation of the sum over all configurations $c$,

$$Z = \sum_c e^{-\beta E(c)} . \quad (275)$$

The quantum version has $\sigma_j$ and $\sigma_k$ replaced by Pauli spin matrices $\sigma^x_j$ and $\sigma^x_k$. The bosons problem is easy, it’s a ferromagnet with all couplings $J_{jk} \geq 0$. The fermion problem has a ”random” sign, which can be mapped on random off-diagonal signs in $J$, so Troyer and Wiese conclude that the sign problem is the origin of the NP hardness. So if $\text{NP} \neq \text{P}$ (solvable in polynomial time): tough luck!
Fermion sign problem; perspectives

Are we out of luck? If we could solve the eigenstates and eigenvalues of \( \hat{H}, \hat{H}|i\rangle = E_i|i\rangle \), we could calculate finite temperature expectation values from the normalized density operator \( \hat{\rho} = \frac{1}{Z} e^{-\beta \hat{H}} \),

\[
\langle \hat{A} \rangle = \text{Tr}(\hat{\rho} \hat{A}) = \frac{\sum_i e^{-\beta E_i} \langle i|\hat{A}|i\rangle}{\sum_i e^{-\beta E_i}} .
\] (276)

Here the weights \( e^{-\beta E_i} \) are positive, so there is no sign problem. Alas, solving \( \hat{H}|i\rangle = E_i|i\rangle \) is an exponentially hard problem. A question one should next ask is, just how explicitly do we need to know the solution to \( \hat{H}|i\rangle = E_i|i\rangle \) in order to avoid the sign problem? Is it enough to have the approximate solution hidden in an evolution algorithm?
Fermion sign problem; perspectives

Chin\textsuperscript{45}: Here we argue that: (1) The sign problem is not intrinsic to solving a fermion problem; (2) it is only a consequence of a poor approximation to the exact propagator; and (3) it can be automatically minimized by using better, higher-order approximate propagators. By following up on the last point, this work shows that by using optimized fourth-order propagators, accurate results can be obtained for up to 20 spin-polarized electrons in a two-dimensional (2D) circular, parabolic quantum dot.

Chin argues that the "sign problem" is due to the large number of antisymmetric free-fermion propagators that are needed to extract the ground state wave function at large imaginary time. \textit{Goal:} Get to a large $\tau$ with less $e^{-\tau \hat{T}}$'s, by using e.g. Eq. (146):

$$\hat{G}_{\text{Chin}}(\tau) = e^{-v_0 \tau \hat{V}} e^{-t_1 \tau \hat{T}} e^{-t_2 \tau \hat{T}} e^{-v_1 \tau \hat{W}} e^{-t_1 \tau \hat{T}} e^{-v_0 \tau \hat{V}}.$$ \hfill (277)

Fermion sign problem; perspectives

Let’s see how the antisymmetric free-fermion propagator looks like. Define, as usual

\[ G_{\hat{T}}(R', R; \tau) = \langle R' | e^{\tau \hat{T}} | R \rangle. \]  

(278)

Antisymmetry comes from the spectral expansion in eigenstates of \( \hat{T} \) with antisymmetric wavefunctions, explicit either as a permutation sum \( \phi_K(R) = \sum_P (-1)^P \phi_{bolzmannon}^{bolzmannon}(\hat{P}R) \) or as a determinant. We know the boltzmannon result, just collect them to a determinant:

\[ G_{\hat{T}}(R', R; \tau) = (4\pi D\tau)^{-3N/2} \det M, \]  

(279)

where the matrix \( M \) has elements

\[ M_{ij} = e^{-\frac{(r_i - r_j')^2}{4D\tau}}. \]  

(280)

This \( \det M \) is where the (almost) random signs come from.
Fermion sign problem; perspectives

The idea is to get the result before the sign problem appears

**FIG. 4.** (Color online) Convergence of the Hamiltonian energy for $N = 20$ polarized electrons using the optimized, fourth-order three- and four-bead propagators. Error bars are computed from 200 to 300 block average of $5 \times 10^4$ configurations of all 20 electrons.
Alternatives with no timestep error: SSE

There are ways to do QMC without any timestep error from splitting $e^{-\beta \hat{H}}$.
One is the **Stochastic Series Expansion (SSE)**. Invented by Anders Sandvik and J. Kurkijärvi \(^{46}\), as an extension to Handscomb’s method. Make a series expansion

$$Z = \text{Tr}[e^{-\beta \hat{H}}] = \sum_{\alpha} \sum_{n=0}^{\infty} \frac{(-\beta)^n}{n!} \langle \alpha | \hat{H}^n | \alpha \rangle$$  \hspace{1cm} (281)

See O. F. Syljyåsen, A. W. Sandvik, Phys. Rev. E 66, 046701 (2002) for **SSE with directed loop update**. SSE has been mostly used in lattice spin systems, such as Ising and Hubbard models.

Alternatives with no timestep error: Continuous time QMC

Continuous time QMC has multiple variants, one way to start is to solve \( \hat{U}(\beta) \) from

\[
e^{-\beta \hat{H}} = e^{-\beta \hat{T}} \hat{U}(\beta) \Leftrightarrow \hat{U}(\beta) = e^{\beta \hat{T}} e^{-\beta \hat{H}}
\]

\[
\Rightarrow \frac{\partial \hat{U}(\beta)}{\partial \beta} = \hat{T} e^{\beta \hat{T}} e^{-\beta \hat{H}} - e^{\beta \hat{T}} \hat{H} e^{-\beta \hat{H}}
\]

\[
= \hat{T} e^{\beta \hat{T}} e^{-\beta \hat{H}} - e^{\beta \hat{T}} (\hat{T} + \hat{V}) e^{-\beta \hat{H}}
\]

\[
= -e^{\beta \hat{T}} \hat{V} e^{-\beta \hat{H}} = -e^{\beta \hat{T}} \hat{V} e^{-\beta \hat{T}} [e^{\beta \hat{T}} e^{-\beta \hat{H}}]
\]

\[
= -e^{\beta \hat{T}} \hat{V} e^{-\beta \hat{T}} \hat{U}(\beta) \equiv -\hat{V}(\beta) \hat{U}(\beta)
\]

and integrate,

\[
\int_0^\beta \frac{\partial \hat{U}(t)}{\partial t} = \hat{U}(\beta) - \hat{U}(0) = -\int_0^\beta dt \hat{V}(t) \hat{U}(t)
\]
Alternatives with no timestep error: Continuous time QMC

Insert and iterate, $\hat{T}_\tau$ is the imaginary-time ordering operator,

$$e^{-\beta \hat{H}} = e^{-\beta \hat{T} \hat{U}(\beta)} = e^{-\beta \hat{T} \left[ 1 - \int_0^\beta d\tau \hat{V}(\tau) \hat{U}(\tau) \right]}$$

$$= e^{-\beta \hat{T} \left[ 1 - \int_0^\beta d\tau_1 \hat{V}(\tau_1) + \int_0^\beta d\tau_1 \int_0^{\tau_1} d\tau_2 \hat{V}(\tau_1) \hat{V}(\tau_2) \ldots \right]}$$

$$= e^{-\beta \hat{T} \left[ 1 - \int_0^\beta d\tau_1 \hat{V}(\tau_1) + \frac{1}{2} \int_0^\beta d\tau_1 \int_0^\beta d\tau_2 \hat{T}_\tau [\hat{V}(\tau_1) \hat{V}(\tau_2)] \ldots \right]}.$$  

This is often written in the form

$$e^{-\beta \hat{H}} = e^{-\beta \hat{T} \hat{T}_\tau \left[ e^{-\int_0^\beta d\tau \hat{V}(\tau)} \right]} , \quad \hat{V}(\tau) \equiv e^{\tau \hat{T}} \hat{V} e^{-\tau \hat{T}} .$$

This can be viewed as the interaction picture representation wrt. $\hat{V}$, and the expanded form does perturbation theory wrt. $\hat{V}$.
Alternatives with no timestep error: Continuous time QMC

Explicitly,

\[
e^{-\beta \hat{H}} = e^{-\beta \hat{T}} - \int_0^\beta d\tau_1 e^{-(\beta - \tau_1) \hat{\tau} \hat{V}} e^{-\tau_1 \hat{T}}
\]

\[
+ \frac{1}{2} \int_0^\beta d\tau_1 d\tau_2 e^{-(\beta - \tau_1) \hat{\tau} \hat{V}} e^{(\tau_1 - \tau_2) \hat{\tau} \hat{V}} e^{-\tau_2 \hat{T}} \ldots .
\]

This gives the canonical partition function, \( \rho_0(R', R; \tau) := \langle R' | e^{-\tau \hat{\tau}} | R \rangle \),

\[
Z = \int dR \langle R | e^{-\beta \hat{H}} | R \rangle = \int dR \rho_0(R, R; \beta)
\]

\[
- \int_0^\beta d\tau_1 \int dR dR_1 \rho_0(R, R_1; \beta - \tau_1) V(R_1) \rho_0(R_1, R; \tau_1)
\]

\[
+ \frac{1}{2} \int_0^\beta d\tau_1 d\tau_2 \int dR dR_1 dR_2 \hat{\tau} \hat{V} \left[ \rho_0(R, R_1; \beta - \tau_1) V(R_1) \rho_0(R_1, R_2; \tau_1 - \tau_2) V(R_2) \rho_0(R_2, R; \tau_2) \right] \ldots .
\]
Alternatives with no timestep error: Continuous time QMC

- This $Z$ is made of DVD-kinks, free diffusion - potential - free diffusion. Free diffusion combined with a singular potential is always bad.

- Terms are positive definite only for an attractive potential – again the Hydrogen atom problem works perfectly. MC sampling of $Z$ could be done by avoiding positive potential regions using domain Green’s functions. Without any tricks the large $|V(R)|$ occasions overwhelm the MC sampling.

Eq. (288) uses $\{R\}$ basis, where $\hat{V}$ is diagonal. You could as well choose an occupation number basis, and divide $\hat{H}$ to diagonal ($\hat{D}$) or off-diagonal ($\hat{Y}$) contributions in this basis. Next, in Eq. (288), replace $\hat{T} \to \hat{Y}$ and $\hat{V} \to \hat{D}$. This leads to CPIMC formulation by Schoof et al. as in Eqs. (92)—(98) in Introduction to Configuration Path Integral Monte Carlo.

---

Things that were not done and things to do

I didn’t talk about

- Path integrals in quantum field theory (Dirac and Majorana fermions etc.), just path integrals in condensed matter physics. If you want to delve into the field theoretical aspects, a good place to start is Edward Witten’s *Fermion Path Integrals And Topological Phases*. A youtube video is also available [here](#).

- Feynman path integrals (see lecture notes, although they are lagging behind these slides)

- Fermion PIMC in detail; I recommend Burkhard Militzer’s excellent presentations on the topic

- Ground state path integrals

Things *you* should do

- find a way around the sign problem