Atomic Scale Simulations

Variance Reduction Techniques

Although the MC error always converges as $N^{-1/2}$ where N is number of sampling points (or equivalently computer time), there are various ways to reduce the coefficient. These techniques go under the name of variance reduction. Consider the integral:

$$I = \int dx f(x) \tag{1}$$

(x could be a multidimensional variable.) Define the efficiency of a MC calculation as:

$$\xi = 1/(t\sigma^2) \tag{2}$$

where t is the time to sample a given point and σ^2 is the variance of a given sample. We want to devise an algorithm to maximize ξ. Usually one does this by reducing the variance. Of course, you can also get a faster computer, or write a more efficient code. It is the product that matters so doing both is even better. The final statistical error ϵ is related to the efficiency as

$$\epsilon = 1/\sqrt{\xi T} \tag{3}$$

where $T = nt$ is the total computer time of run.

Importance sampling

The most common MC variance reduction method is to optimize the sampling of x. Suppose we sample x from $p(x)$ where $p(x)$ is arbitrary for the moment. Then we rewrite the integral as:

$$I = \int dx p(x)[f(x)/p(x)] = \langle f(x)/p(x)\rangle_p \tag{4}$$

where the subscript indicates the sampling function. The *estimator* is $f(x)/p(x)$. The variance is:

$$\sigma^2 = \langle [f(x)/p(x) - I]^2\rangle = \int dx f(x)^2/p(x) - I^2. \tag{5}$$

The mean value of the estimator is independent of $p(x)$ but the variance is not. To simplify things, let us assume that the CPU time/sample (t) is independent of $p(x)$. Then we can vary $p(x)$ to minimize σ^2. Remember that $p(x)$ must remain a probability distribution. The simplest way to maintain the constraints is to substitute $p(x) = q^2(x)/\int dx q^2(x)$ where $q(x)$ is a completely arbitrary real function. After solving for $\delta\sigma^2/\delta q(x) = 0$ we find the optimal sampling distribution:

$$p^*(x) = \frac{|f(x)|}{\int dx |f(x)|}. \tag{6}$$

The optimal estimator is:

$$f(x)/p^*(x) = sign(f(x)) \int dy |f(y)| \tag{7}$$

Our estimator is $\pm constant$; it only changes when the sign of $f(x)$ changes!

We see that if f is entirely positive or negative, the optimal variance is *zero*. This is an example of a *zero variance principle*. As the sampling approaches the optimal sampling, the variance approaches zero. But notice that we can only obtain the optimal by performing the integral since we need I to be able to sample $p^*(x)$. This is typical. Optimal sampling requires first solving the problem! However, by using some improved sampling, incorporating some of the features of p^*, one can drastically reduce the variance.

Example: let $f(x) = \exp(-x^2/2)/(1+x^2)$ be integrated from $(-\infty$ to $\infty)$. Let us take as an importance sampling function $p(x) = (2\pi a)^{-1/2} exp(-x^2/(2a))$ with a a free parameter. Numerically we find the minimum occurs at $a \approx 0.6$ getting a reduction of 9 in the variance over the "obvious" value of $a = 1$. The estimator is proportional to $\exp(-1/2(1-1/a))/(1+x^2)$ and hence blows up at large x if $a < 1$. But the variance integral goes as $\exp(-1/2(2-1/a))$. It exists only for $a > 1/2$! The figures show what happens when you go ahead and importance sample in such a way that the variance does not exist.

The code that produced these numbers is:

```
      nsamp=10000 ! number of samples
      a=0.6       ! sampling parameter
! some constants
      sa=sqrt(a)
      pi=3.14159265
      s2pia=sqrt(2*pi*a)
      exa=(1/a-1)/2
! zero mean and variance
      xm=0
      x2=0
      do isamp=1, nsamp ! loop over samples
          chi=sqrt(-2*log(ranf())*cos(pi*ranf()) ! compute ndrn
          x=sa*chi            ! scale by a
          e=s2pia*exp(x*x*exa)/(x*x+1) !estimator
          xm=xm+e ! add to mean
          x2=x2+e**2 ! add to square
      enddo
      write (*,*) 'mean =',xm/nsamp,
     +' error=',sqrt((x2/nsamp-(xm/nsamp)**2)/(nsamp-1))
      end
```

Important rules on setting up an importance sampling:

1. Once you have chosen $p(x)$, prove that the variance exists. Look at the limits at large x and at places where $p(x)$ or $f(x)$ vanish. Look at the distribution of values to make sure it is Gaussian.

2. In general, do not under sample. Make sure that $f(x)/p(x)$ never gets too big. In the above example, increasing a is stable, decreasing a is dangerous.

3. The value, with the computed errors, should be independent of $p(x)$. Make sure you pass this test. Otherwise you have a bug.

4. Once you are close to the minimum variance, just run longer. You can waste a lot of time trying to further optimize.

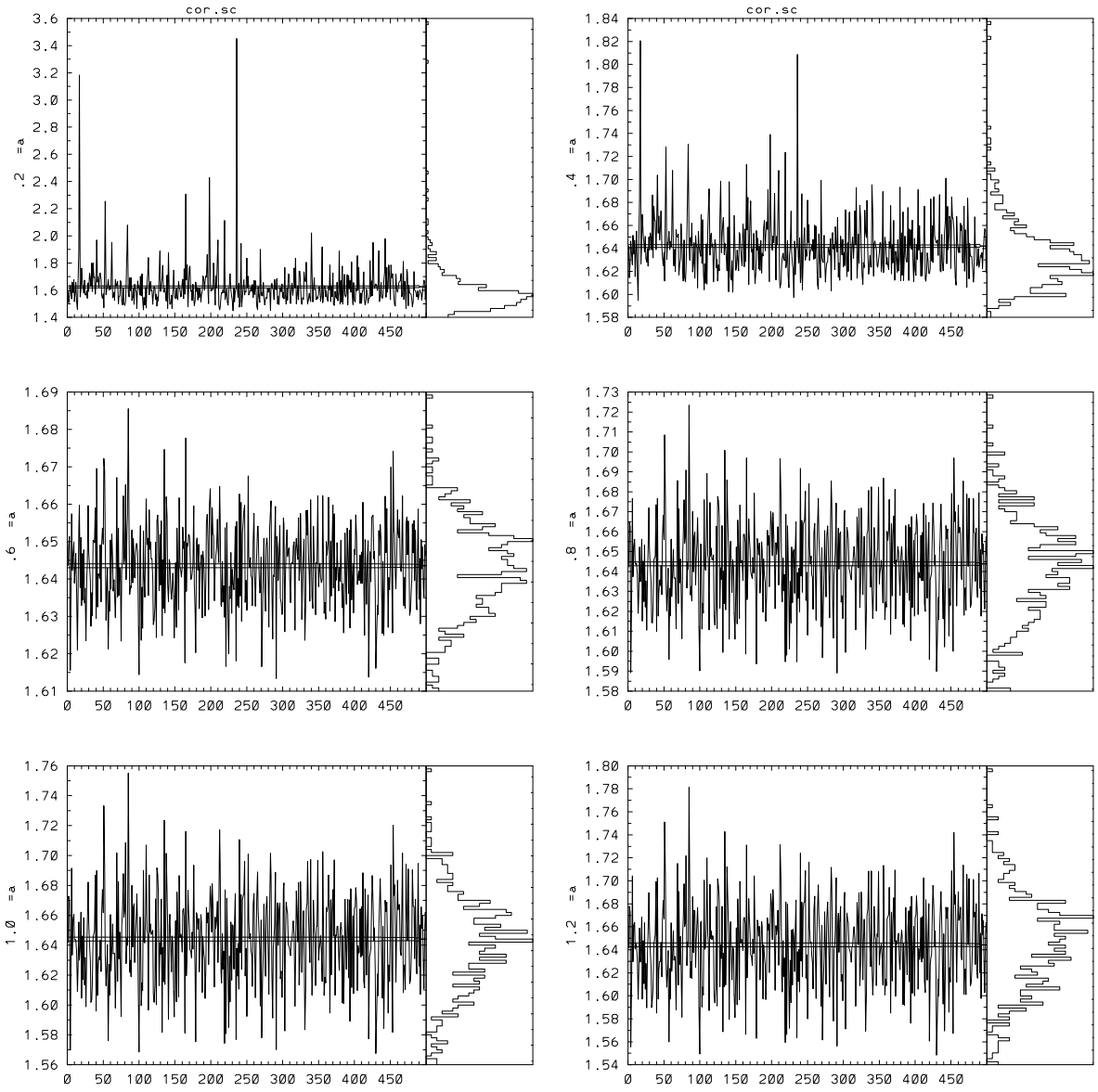

Figure 1: A plot of the estimate versus block (each block is 200 steps) of the example for various values of α. From left to right, top to bottom, $a = .2, .4, .6, .8, 1.0, 1.2$. Spikes show non-Gaussian behavior.

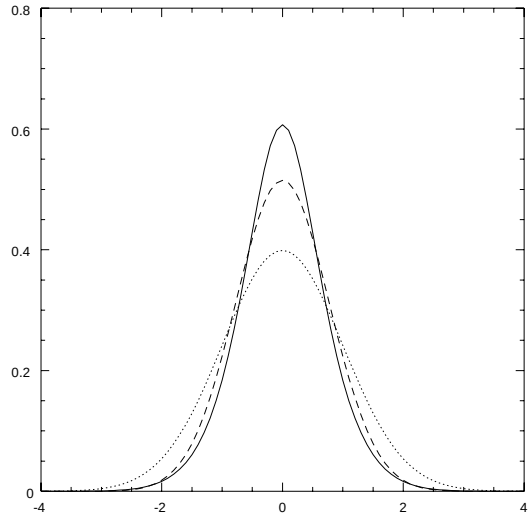

Figure 2: The optimal importance function (solid line) versus the simple one ($\alpha = 1$, dotted line) and the optimized gaussian $\alpha = 0.6$, dashed line).

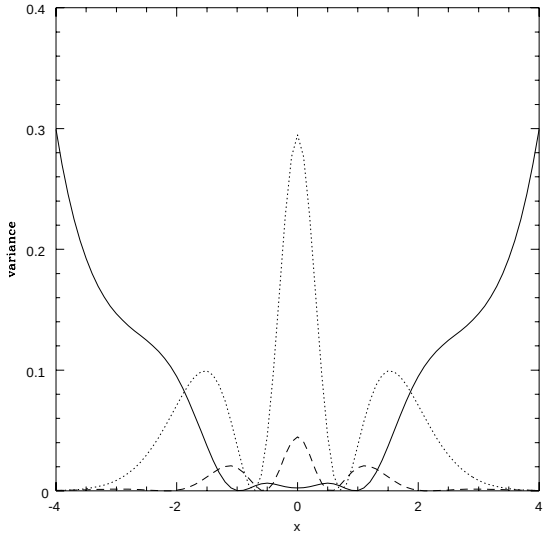

Figure 3: The variance integrand for $\alpha = 1$ (dotted line), the optimized gaussian $\alpha = 0.6$ (dashed line), and for an infinite variance value $\alpha = 0.4$ (solid line).

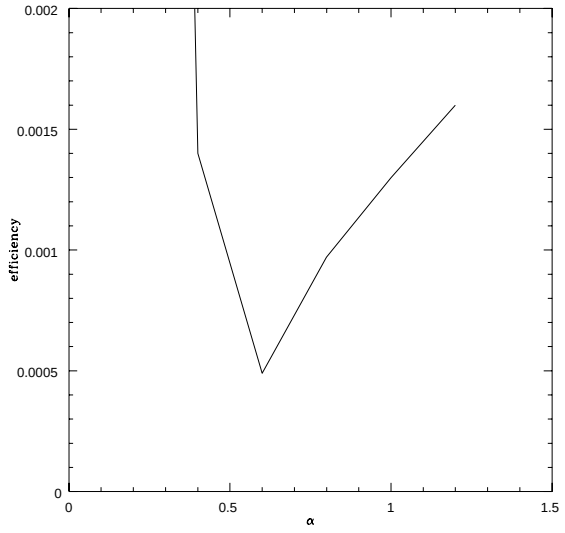

Figure 4: The overall efficiency as a function of α.

For functions with both positive and negative pieces, optimal sampling does not lead to zero variance. Let us define the fraction of time that one samples positive (or negative) values of f as:

$$n_+ = \int dx|f(x)|\theta(f(x) > 0)/Z \tag{8}$$

with a similar definition for n_- and $Z = \int dx|f(x)|$. Clearly: $I = Z(n_+ - n_-)$. The quantity in parentheses is known as the "average sign". It can be shown by rearranging terms that the minimum variance is:

$$\sigma^2 = Z^2(1 - (n_+ - n_-)^2) = 4Z^2 n_+ n_- \tag{9}$$

Hence we want to make sure that our integral is either mostly positive or mostly negative, otherwise there is a considerable loss of efficiency. The relative error is:

$$\sigma/I = \sqrt{(n_+ - n_-)^{-2} - 1} \tag{10}$$

which becomes very large if the average sign is close to zero. This is why we say: *Monte Carlo is good at adding but cannot subtract.* One might think that one could simply shift the function by adding a constant, but this rarely works very well, particularly in high dimensions or for integrals over infinite domains.

Antithetic Variates

The idea here is to use negative correlation to reduce the variance. That is we couple a point where f is large (x such that $f(x)/p(x) > I$) with a point f is small. Suppose f is generally positive for $x > 0$ and generally negative for $x < 0$. Then we write the integral

$$I = \int_0^\infty dx[f(x) + f(-x)] \tag{11}$$

We couple the evaluation of x with $-x$. With the assumed properties, the variance of $\hat{f}(x) = 1/2[f(x) + f(-x)]$ may be much less than the variance of $f(x)$. Also $\hat{f}$ may be more suitable for the application of importance sampling.

Correlated sampling

Suppose what we really want is not a single integral but some function of several integrals. Let

$$F_k = \int dx f_k(x) \tag{12}$$

and what we want to calculate is $G(F_1, F_2 \ldots)$. A common example (which is treated in homework) is $G = F_1/F_2$. The straightforward way of doing this is to run independent MC evaluations of $F_1, F_2 \ldots$ then combine the final results to get a good estimate of G. The idea of correlated sampling is to use the same points $\{x_i\}$ to estimate all the integrals F_k at once. (One can either do the integrals altogether on the code or simply to the calculation again with the same random number sequence. It is just a question of convenience versus speed.)

A particularly favorable situation is in estimating a derivative. Suppose we want to calculate the derivative of an integrand with respect to some parameter (say a). One can either formally carry through the derivative and then analyze how to best sample the resulting integrand, or one can simply run the code twice with the parameter having two different but nearby values, but using the same random number sequence.

The homework exercise is to develop an expression for the variance of G, then to optimize $p(x)$ to minimize it. The optimal importance function is given by:

$$p^*(x) \propto \left| \sum_{k=1}^{N} f_k(x) \frac{dG}{dF_k} \right| . \tag{13}$$

Let us mention a related problem, namely for any non-linear function G, one will bias the result. This means that

$$< G >_N \neq G(< F_1 >_N, < F_2 >_N \ldots) \tag{14}$$

where the average is over samples of size N. Our estimate of G may be systematically high or low. To get an estimate of this bias, let us assume we can do a Taylor expansion of G about the converged value.

$$< G >_N - G(< F_1 >_N, < F_2 >_N \ldots) = \sum_{i,j} \frac{d^2}{dF_i dF_j} G(F_1, F_2 \ldots)|_{<F_1>\ldots} covar(F_1, F_2) \tag{15}$$

Note that the bias is always order N^{-1} so for large number of samplings it should be smaller than the statistical error which is order $N^{-1/2}$.

A disadvantage of correlated sampling is that the importance function for minimizing the variance of G may be bad for an individual f_k. It may be better to sample separately, particularly when the integrals have little in common.

Hammersley and Handscomb, pgs 50-75.
Kalos and Whitlock, pgs 89-114.