

MCASE

Kevin Ly

February 18, 2025

The purpose of MCASE is to evaluate the following integrals using a Metropolis-Hastings algorithm:

$$\begin{aligned}\langle A \rangle_{\text{NVT}} &= \frac{\int d\{\mathbf{R}\} A(\{\mathbf{R}\}) \exp[-\beta U(\{\mathbf{R}\})]}{\int d\{\mathbf{R}\} \exp[-\beta U(\{\mathbf{R}\})]} \\ \langle A \rangle_{\text{NPT}} &= \frac{\int dV d\{\mathbf{s}\} V^N A(\{\mathbf{R}\}) \exp(-\beta[U + PV])}{\int dV d\{\mathbf{s}\} V^N \exp(-\beta[U + PV])}\end{aligned}\tag{1}$$

where A is an observable of interest, V is the volume of the simulation cell, and $\{\mathbf{s}\}$ is the set of scaled coordinates. The quantum estimates are obtained by simply replacing the potential with an effective potential $U \rightarrow \tilde{U}$ according to the isomorphism attained through the path integral formalism.

I will begin by describing how to draw samples from the relative distribution $P(\{\mathbf{R}\}) = e^{-\beta U(\{\mathbf{R}\})}$ (NVT); the routines for sampling the other distributions (to be described) will ultimately be based on this scheme. A trajectory of configurations is produced by using the *all-particle* move

$$\begin{aligned}T(\{\mathbf{R}\} \rightarrow \{\mathbf{R}'\}) &\propto \prod_i \exp\left(-\frac{(\mathbf{R}'_i - \mathbf{R}_i - \tau\beta\mathbf{F}_i)^2}{4\tau}\right) \\ &= \exp\left(-\sum_i \frac{(\mathbf{R}'_i - \mathbf{R}_i - \tau\beta\mathbf{F}_i)^2}{4\tau}\right)\end{aligned}\tag{2}$$

where τ is a user-specified step size, and \mathbf{F}_i is the force on particle i in the *current configuration* $\{\mathbf{R}\}$. The proportionality in the first line is to indicate that there is an (irrelevant) normalization factor in front. The reverse probability $T(\{\mathbf{R}'\} \rightarrow \{\mathbf{R}\})$, which is required for calculating the acceptance ratio, involves the forces calculated in the proposed configuration $\{\mathbf{R}'\}$.

This is a product of Gaussians, each centered at $\mathbf{R}_i + \tau\beta\mathbf{F}_i$, with width

$\sigma = \sqrt{2\tau}$. Therefore, to execute this move in practice, every coordinate is moved by adding $\boldsymbol{\chi} + \tau\beta\mathbf{F}$, where $\boldsymbol{\chi}$ is a random vector whose components are independently drawn from the normal distribution with $\sigma = \sqrt{2\tau}$. While it is intuitive to push the particle in the direction of the force exerted upon it, the formal reason for doing so is because this is the direction of increasing probability:

$$\begin{aligned}\nabla_i e^{-\beta U(\{\mathbf{R}\})} &= (-\beta \nabla_i U) e^{-\beta U} \\ \implies \nabla_i \log P &= -\beta \mathbf{F}_i\end{aligned}\tag{3}$$

where, by definition, $\mathbf{F}_i = -\nabla_i U$. The direction of increasing log probability is also the direction of increasing probability.

To sample the relative distribution $e^{-\beta(U+PV)+N\log V}$, a trajectory is produced by alternating between the move (2) and a move which changes the shape of the simulation cell. In MCASE, the shape of the cell is a parallelepiped characterized by a vector \mathbf{c} of six numbers: the three side lengths, and the three angles between the sides. The “cell move” is

$$T(\mathbf{c} \rightarrow \mathbf{c}') \propto \begin{cases} 1, & \mathbf{c}' - \mathbf{c} \in [-\sigma_l, \sigma_l]^3 \times [-\sigma_a, \sigma_a]^3 \\ 0, & \text{otherwise.} \end{cases}\tag{4}$$

where σ_l is a user-specified step size for the side lengths, and σ_a for the angles. This is a uniform distribution over the hyperrectangle centered at \mathbf{c} . To carry out the move, $\sigma_l\chi$ is added to each side length, and $\sigma_a\chi$ is added to each angle, where χ is uniformly drawn from $[-1, 1]$; an independently drawn χ is used for each component. Unlike the particle move, (4) is symmetric $T(\mathbf{c} \rightarrow \mathbf{c}') = T(\mathbf{c}' \rightarrow \mathbf{c})$. As a result, the acceptance ratio for this move simplifies to

$$\mathcal{A}(\mathbf{c} \rightarrow \mathbf{c}') = \exp[-\beta(P[V' - V] + [U' - U]) + N \log(V'/V)].\tag{5}$$

When changing the simulation cell, the scaled coordinates are kept fixed, meaning that the real coordinates are changed. As a result, the change in energy is generally non-zero, and must be computed for this step. However, no derivatives are required.

Allowing arbitrary changes to the shape of the cell is important when simulating solids, but can be detrimental for simulating liquids. This is because liquids can freely conform to their containers. This means that, at a given

pressure and temperature, although the cell volume will fluctuate about some equilibrium value, the shape can be arbitrarily deformed. To prevent this, there is a separate algorithm in which a cubic cell, which is characterized by only a single side length L , will remain cubic. Changes to the cell occur only through changing this one side length, rather than all 6 cell parameters.

To recap, the main loop in the classical algorithm is

1. Propose new coordinates for every particle $\mathbf{R}' = \mathbf{R} + \tau\beta\mathbf{F} + \boldsymbol{\chi}$.
2. Compute the new energy and new forces for the proposed configuration using the ASE calculator of choice.
3. Compute the acceptance ratio

$$\mathcal{A}_p = \exp [\log T - \log T' - \beta\Delta U] \quad (6)$$

where T is (2) and T' is the reverse probability. Even when we are sampling the NPT ensemble, the volume terms do not enter here because the cell is not being changed in this move.

4. Accept or reject: either all quantities are updated (accepted) or reverted (rejected).
- (NPT) 5. Propose a new cell shape by adding to each side length a uniformly drawn random number from $[-\sigma_l, \sigma_l]$, and adding to each angle a uniformly drawn random number from $[-\sigma_a, \sigma_a]$.
- (NPT) 6. Compute the new energy of the proposed configuration. The new forces are not required at this step, so that if this cell move is rejected, this skips an excess force calculation.
- (NPT) 7. Compute the acceptance ratio (5).
- (NPT) 8. Accept or reject: either all quantities are updated, requiring the force calculation skipped earlier, or reverted.

Conveniently, the NVT and NPT algorithms are identical up to the last 4 steps.

Path-integral simulations in MCASE follow the same algorithm just described but with the effective potential instead of U . For convenience, the effective

potential is reproduced below:

$$\tilde{U} = \frac{pm}{2\beta^2\hbar^2} \sum_{i=1}^p \sum_{j=1}^N |\mathbf{R}_j^i - \mathbf{R}_j^{i+1}|^2 + \frac{1}{p} \sum_{i=1}^p U(\{\mathbf{R}\}_i) \quad (7)$$

where p is the number of beads in the path, m is the mass of the particles (assumed to all be identical), $\{\mathbf{R}\}_i$ is the i th bead in the path, and \mathbf{R}_j^i is the coordinate of particle j in bead i . At any given step, the state of the system is characterized not by a single configuration of coordinates, but rather a set of p configurations. Each configuration is kept as its own ASE object, so that total energies and forces are calculated on a per-bead basis. For this reason, only one bead is moved at a time, though all particles within that bead will be moved.

For simulations at fixed pressure, the simulation cell is identical throughout the entire path. As a result, whenever a change in simulation cell is proposed, this change will apply to all beads. In MCASE, all-particle moves will be proposed for every bead before a cell move is proposed. This means that, for a path with p beads, there will be p all-particle moves for every cell move.