

Constant-pressure nested sampling with atomistic dynamicsRobert J. N. Baldock,^{1,*} Noam Bernstein,² K. Michael Salerno,³ Lívia B. Pártay,⁴ and Gábor Csányi⁵¹*Theory and Simulation of Materials (THEOS), and National Centre for Computational Design and Discovery of Novel Materials (MARVEL), École Polytechnique Fédérale de Lausanne, CH-1015 Lausanne, Switzerland*²*Center for Materials Physics and Technology, Naval Research Laboratory, Washington, DC 20375, USA*³*National Research Council Associateship Program, resident at the US Naval Research Laboratory, Washington, DC 20375, USA*⁴*Department of Chemistry, University of Reading, Reading, United Kingdom*⁵*Engineering Laboratory, University of Cambridge, Cambridge, United Kingdom*

(Received 18 April 2017; revised manuscript received 7 September 2017; published 30 October 2017)

The nested sampling algorithm has been shown to be a general method for calculating the pressure-temperature-composition phase diagrams of materials. While the previous implementation used single-particle Monte Carlo moves, these are inefficient for condensed systems with general interactions where single-particle moves cannot be evaluated faster than the energy of the whole system. Here we enhance the method by using all-particle moves: either Galilean Monte Carlo or the total enthalpy Hamiltonian Monte Carlo algorithm, introduced in this paper. We show that these algorithms enable the determination of phase transition temperatures with equivalent accuracy to the previous method at $1/N$ of the cost for an N -particle system with general interactions, or at equal cost when single-particle moves can be done in $1/N$ of the cost of a full N -particle energy evaluation. We demonstrate this speed-up for the freezing and condensation transitions of the Lennard-Jones system and show the utility of the algorithms by calculating the order-disorder phase transition of a binary Lennard-Jones model alloy, the eutectic of copper-gold, the density anomaly of water, and the condensation and solidification of bead-spring polymers. The nested sampling method with all three algorithms is implemented in the PYMATNEST software.

DOI: [10.1103/PhysRevE.96.043311](https://doi.org/10.1103/PhysRevE.96.043311)**I. INTRODUCTION**

The ability to predict the behavior of materials under a variety of conditions is important in both academic and industrial settings. In principle, statistical mechanics enables the prediction of the properties of materials in thermodynamic equilibrium from the microscopic interaction of atoms. Computer simulation is the tool for using numerical statistical mechanics in practice, and a wide variety of models and approximations are used for atomic interactions, all the way from the electronic Schrödinger equation to simple hard spheres. The equilibrium pressure-temperature phase diagram for a given composition is one of the most fundamental properties of a material, and forms the basis for making, changing, designing, or in general, just thinking about the material.

The common approach is to use different computational methods for resolving each of the transitions between the phases, or for comparing the stability of particular combinations of phases. This requires the prior knowledge of a list of proposed phases or crystal structures at each set of thermodynamic parameters. In a previous paper [1] we introduced a nested sampling (NS) algorithm [2,3] that enables the automated calculation of complete pressure-temperature-composition phase diagrams. The nested sampling algorithm constitutes a single method for resolving all phase transitions automatically in the sense that no prior knowledge of the phases is required.

The algorithm in Ref. [1] used Gibbs sampling, i.e., single-particle Monte Carlo (SP-MC) moves, to explore configuration space. For some interactions, which we term *separable*, the

energy change of an N -particle system due to the displacement of a single particle can be calculated in $1/N$ times the cost of evaluating the energy of the whole system. Thus, for separable interactions, the cost of an N -particle sweep is equal to the cost of a single full-system energy/force evaluation. For these cases Gibbs sampling is efficient. However, interactions in general are not separable, and if a single-particle move is just as costly as a full-system evaluation, a sweep that moves every particle costs N times more than a single full-system energy/force calculation. All-particle MC moves can be used in principle, but it is well known that making such moves in random directions leads to very slow exploration in condensed phases (liquids, solids), because maintaining a finite MC acceptance rate requires that the displacement of each atom become smaller as the system size increases [4].

Here we replace Gibbs sampling by either one of two algorithms that use efficient all-particle moves inspired by the Hamiltonian Monte Carlo (HMC) method [5]. The purpose of this paper is to show how these algorithms can be utilized efficiently in the case of nested sampling. One is a total enthalpy HMC (TE-HMC) introduced in this paper, which takes advantage of standard molecular dynamics (MD) implemented in many simulation packages. The other is Galilean Monte Carlo (GMC) [6,7], which is not as widely available, but does not suffer from the breakdown of ergodicity that the TE-HMC algorithm may experience for systems with a large number of particles. We compare the efficiency of these two methods with that of Gibbs sampling, and find that they require comparable numbers of whole-system energy/force evaluations, leading to a factor of $1/N$ reduction in computational time for general inter-particle interactions.

The GMC and TE-HMC algorithms use interparticle force information to move all particles coherently along “soft”

*rjnbaldock@gmail.com

degrees of freedom, and therefore explore faster than simple diffusion, at least over short time scales. Effectively, short trajectories are all-particle MC move proposals with large step lengths that still lead to reasonable MC acceptance probabilities. Over the time scale of an entire Markov chain Monte Carlo (MCMC) walk, the motion is still diffusive, but the short-time coherence helps GMC and TE-HMC explore configuration space much faster than randomly oriented all-particle moves. Using GMC or TE-HMC enables the simulation of a wide range of phase transitions in atomistic and particle systems with NS, including chemical ordering in binary Lennard-Jones (LJ), the eutectic of copper-gold alloys, freezing of water, and the transitions of a coarse-grained bead-spring polymer model. Parallel implementations of both algorithms are available in the PYMATNEST PYTHON software package [8], using the LAMMPS package [9] for the dynamics.

II. THE NESTED SAMPLING METHOD

The nested sampling algorithm with constant pressure and flexible boundary conditions (i.e., with variable periodic cell shape and volume) calculates the cumulative density of states, $\chi(\tilde{H})$, at fixed pressure P , where $\tilde{H} = U + PV$ is the configurational enthalpy, $U(\mathbf{r})$ is the potential energy function, and V is the volume of the system. From the cumulative density of states, one can calculate the partition function and heat capacity as explicit functions of temperature. Nested sampling also returns a series of atomic configurations, from which one may compute ensemble averages of observables and free energy landscapes [1].

The key idea of nested sampling is that it constructs a series of decreasing enthalpy levels $\{\tilde{H}_i^{\text{sup}}\}$, each of which bounds from above a volume of configuration space χ_i , with the property that χ_i is approximately a constant factor smaller than the volume χ_{i-1} of the level above. The constant-pressure partition function and its approximation in nested sampling are then given by

$$\Delta(N, P, \beta) = \frac{\beta P}{N!} \left(\frac{2\pi m}{\beta h^2} \right)^{3N/2} \int_{-\infty}^{\infty} d\tilde{H} \frac{\partial \chi}{\partial \tilde{H}} e^{-\beta \tilde{H}} \quad (1)$$

$$\approx \frac{\beta P}{N!} \left(\frac{2\pi m}{\beta h^2} \right)^{3N/2} \sum_i (\chi_{i-1} - \chi_i) e^{-\beta \tilde{H}_i^{\text{sup}}}. \quad (2)$$

Here N is the number of particles of mass m , h is Planck's constant, and $\partial \chi / \partial \tilde{H}$ is the density of enthalpy states.

To calculate the absolute value of the partition function (2), we must possess the absolute values of the configuration space volumes $\{\chi_i\}$. The volumes $\{\chi_i\}$ are specified in NS as a decreasing geometric progression, starting from χ_0 , which is the total (finite) volume of the configuration space. The configuration space must therefore be compact. In order to ensure that we are sampling from a compact configuration space, the simulation cell volume V is restricted to be less than a maximum value V_0 , chosen to be sufficiently large as to correspond to an almost ideal gas. The total configuration space volume is therefore $\chi_0 = V_0^{N+1} / (N+1)$. Restricting the sampling to $V < V_0$ allows a good approximation of the partition function provided $k_B T \ll P V_0$ [1].

The simulation cell is periodic, and is represented by \mathbf{h} , a 3×3 matrix of lattice vectors that relates the Cartesian

positions of the particles \mathbf{r} to the fractional coordinates \mathbf{s} by $\mathbf{r} = \mathbf{h}\mathbf{s}$. The volume of the simulation cell is $V = \det \mathbf{h}$, and $\mathbf{h}_0 = \mathbf{h}V^{-1/3}$ is the image of the unit cell normalized to unit volume. The NS algorithm maintains a pool of K configurations drawn from

$$\text{Prob}(\mathbf{s}, \mathbf{h}_0, V | V_0, \tilde{H}^{\text{sup}}, d_0) \propto V^N \delta(\det \mathbf{h}_0 - 1) \Theta(\tilde{H}^{\text{sup}} - \tilde{H}) \Theta(V_0 - V) \prod_{i=1}^{3N} \Theta(s_i) \Theta(1 - s_i) \Theta\left(d_0 - \min_{j \neq k} \left[\frac{1}{\mathbf{h}_0^j \times \mathbf{h}_0^k} \right]\right), \quad (3)$$

where Θ is the Heaviside step function, \mathbf{h}_0^i are columns of \mathbf{h}_0 , and \tilde{H}^{sup} is a maximum configurational enthalpy. Thus, fractional particle coordinates \mathbf{s} are uniformly distributed on $(0,1)^{3N}$, \tilde{H} is restricted to be smaller than \tilde{H}^{sup} , and V is restricted to be smaller than V_0 . The last term in Eq. (3) restricts the simulation cell from becoming too thin (controlled by the parameter d_0), thus avoiding unphysical correlations between interacting periodic images [1]. For simple fluids of 64 atoms, d_0 should be set no smaller than 0.65, while simulations with larger numbers of atoms can tolerate smaller values of d_0 [1] (see Appendix E). The probability distribution (3) corresponds to a uniform distribution over the Cartesian particle coordinates \mathbf{r} , subject to the constraints above.

The simulation is initialized by drawing K configurations from distribution (3), with $\tilde{H}^{\text{sup}} = \infty$. After initialization, the NS algorithm performs the following loop, starting with $i = 1$:

(1) From the set of K configurations, record the $K_r \geq 1$ samples with the highest configurational enthalpy, $\{\tilde{H}\}_i$. Use the lowest enthalpy from that set, $\min\{\tilde{H}\}_i$, as the new enthalpy limit: $\tilde{H}^{\text{sup}} \leftarrow \min\{\tilde{H}\}_i$. The volume of configuration space with enthalpy equal to or less than \tilde{H}^{sup} is $\chi_i \approx \chi_0 [(K - K_r + 1)/(K + 1)]^i$.

(2) Remove the K_r samples with enthalpies $\{\tilde{H}\}_i$ from the pool of samples and generate K_r new configurations from the distribution (3), using the updated value of \tilde{H}^{sup} . This is achieved by first choosing K_r random configurations from the pool of remaining samples, creating clones of those configurations, and evolving the cloned configurations using a MCMC algorithm that converges to the distribution (3).

(3) Set $i \leftarrow i + 1$, and return to step 1 unless a stopping criterion is met (see Appendix A).

The required values of K and the MCMC walk length, L , depend on the system being studied. This behavior is described in Refs. [1–3,10]. In addition, increasing K_r allows for greater parallelization of the NS algorithm; however, increasing $\frac{K_r-1}{K}$ also leads to greater error in the estimates of $\{\chi_i\}$, so care should be taken not to make $\frac{K_r-1}{K}$ too large (see Appendix B).

Thermodynamic expectation values and free energy landscapes can be computed using the samples recorded during step 1 of the NS algorithm [1]. Representative configurations can be sampled at any temperature simply by choosing configurations at random, according to their thermalized probabilities

$$p_i(\beta) = \frac{(\chi_{i-1} - \chi_i) e^{-\beta \tilde{H}_i}}{\sum_i (\chi_{i-1} - \chi_i) e^{-\beta \tilde{H}_i}}. \quad (4)$$

Examining a small number of configurations chosen in this way is often sufficient to understand which phase occurs at each temperature. This method was used to choose the atomic configurations shown in Sec. IV.

III. MARKOV CHAIN MONTE CARLO ALGORITHMS

To decorrelate the cloned configurations in step 2 of the NS algorithm we use a MCMC algorithm that converges to the distribution (3) by applying two kinds of steps: cell steps, including changes to volume and shape, and particle steps, including continuous motion in space and (for multicomponent systems) coordinate swaps between particles of different types.

The cell steps include volume steps that ensure $\text{Prob}(V) \propto V^N$, and shearing and stretching steps that lead to $\text{Prob}(\mathbf{h}_0) \propto \delta(\det \mathbf{h}_0 - 1) \Theta(d_0 - \min_{i \neq j} [\frac{1}{\mathbf{h}_0^i \times \mathbf{h}_0^j}])$, as required by the target distribution [1]. The following subsections introduce the algorithms for moving the configuration in the space of the atomic coordinates.

A. Galilean Monte Carlo

In GMC [6,7] one defines an infinite square-well potential function \tilde{H}_{GMC} ,

$$\tilde{H}_{\text{GMC}}(\mathbf{s}, V, \mathbf{h}_0) = \begin{cases} 0, & \tilde{H} < \tilde{H}^{\text{sup}}, \\ \infty, & \tilde{H} \geq \tilde{H}^{\text{sup}}, \end{cases} \quad (5)$$

which is equal to the logarithm of the desired probability distribution: uniform over the allowed region, $\tilde{H} < \tilde{H}^{\text{sup}}$, and zero elsewhere. Note that we have omitted the constraints on V and \mathbf{h}_0 from (5) since we use GMC to explore only the atomic coordinates. Having defined \tilde{H}_{GMC} , one samples the fractional atomic coordinates \mathbf{s} uniformly by performing standard Hamiltonian Monte Carlo sampling [5] on the function \tilde{H}_{GMC} . We follow the GMC approach proposed by Betancourt [7], which uses a fixed number of force evaluations and therefore helps the load balance when parallelizing the algorithm (see Appendix B). Our implementation, expressed for practical convenience of implementation in Cartesian coordinates \mathbf{r} rather than fractional coordinates \mathbf{s} , is as follows.

At the start of each atomic GMC trajectory, save the initial atomic coordinates \mathbf{r}_0 . Generate a velocity \mathbf{v} chosen uniformly from the surface of a $3N$ -dimensional hypersphere of radius 1. Repeat the following loop L times:

(1) Propagate the atomic coordinates \mathbf{r} in the direction of \mathbf{v} for one step of length dt . The atomic coordinates are now \mathbf{r}^* .

(2) If $\tilde{H} \geq \tilde{H}^{\text{sup}}$, attempt to redirect the trajectory back into the allowed region by reflecting velocities from the current position, by $\mathbf{v} \leftarrow \mathbf{v} - 2(\mathbf{v} \cdot \hat{\mathbf{n}})\hat{\mathbf{n}}$, where $\hat{\mathbf{n}} = -\nabla_{\mathbf{r}} \tilde{H} / |\nabla_{\mathbf{r}} \tilde{H}|$. Following velocity reflection, propagation continues from \mathbf{r}^* .

Finally, if at the end of the trajectory $\tilde{H} \geq \tilde{H}^{\text{sup}}$, reject the trajectory and return to the initial atomic coordinates \mathbf{r}_0 . Note that the reflection in step 2 *does not* occur at the exact boundary $\tilde{H} = \tilde{H}^{\text{sup}}$, and this is essential to maintain detailed balance. The acceptance rate of the GMC step is controlled by adjusting dt . There are similarities between GMC and the ‘‘hit and run’’ algorithm for sampling convex volumes [11], and it remains to be seen whether the clear advantages over Gibbs sampling have common underlying reasons [12].

B. Total enthalpy Hamiltonian Monte Carlo

One disadvantage of GMC is that when the boundary of the allowed region of configuration space $\tilde{H} < \tilde{H}^{\text{sup}}$ is complicated, attempts to reflect the sampler back into the allowed region fail frequently, and reflection further into the disallowed region often leads to rejection of the entire trajectory, thus overall driving down the optimal step size. Hamiltonian (constant-energy) molecular dynamics, on the other hand, can generate nearly constant *total* energy trajectories using comparatively large step sizes, and use the exchange of energy between potential and kinetic degrees of freedom to smooth the ‘‘reflections’’ from high potential energy regions. In TE-HMC we take advantage of this behavior by using short MD trajectories to evolve the atomic coordinates.

Hamiltonian dynamics couples the evolution of the atomic momenta and coordinates, and in TE-HMC we explicitly sample the *total phase space* of the atoms $(\mathbf{s}, V, \mathbf{h}_0, \mathbf{p})$, where \mathbf{p} denotes the Cartesian momenta. In contrast, in SP-MC and GMC one samples only the atomic *configuration* space $(\mathbf{s}, V, \mathbf{h}_0)$.

In step 1 of each NS iteration the K_r samples with highest total enthalpy $H = (\tilde{H} + E_k)$, where $E_k(\mathbf{p})$ is the kinetic energy, are identified as the next set of recorded samples. Next, the total enthalpy limit is updated $H^{\text{sup}} \leftarrow \min\{H\}_i$, and in step 2 K_r new samples are generated from the joint probability distribution

$$\begin{aligned} & \text{Prob}(\mathbf{s}, \mathbf{h}_0, V, \mathbf{p} | V_0, H^{\text{sup}}, d_0, E_k^0) \\ & \propto V^N \delta(\det \mathbf{h}_0 - 1) \Theta(H^{\text{sup}} - H) \Theta(V_0 - V) \\ & \Theta(E_k^0 - E_k(\mathbf{p})) \prod_{i=1}^{3N} \Theta(s_i) \Theta(1 - s_i) \\ & \Theta\left(d_0 - \min_{j \neq k} \left[\frac{1}{\mathbf{h}_0^j \times \mathbf{h}_0^k} \right] \right). \end{aligned} \quad (6)$$

Distribution (6) invokes the same constraints on V, \mathbf{h}_0 as distribution (3), but restricts $H < H^{\text{sup}}$ and specifies that the momenta are uniformly distributed in the region

$$E_k(\mathbf{p}) < E_k^0. \quad (7)$$

Using both maximum volume and kinetic energy values V_0 and E_k^0 ensures that the phase space we sample is compact, the necessity of which is explained in Sec. II. In particular, V_0 and E_k^0 enforce compactness of the sampled configuration and momentum spaces, respectively.

We initialize exactly as described in Sec. II, except that we now assign each sample momenta chosen uniformly at random from the region (7). This is achieved using Algorithm 2 given in Appendix C. We choose $E_k^0 = \frac{3}{2} P V_0$ as for an ideal gas, so that, again, we obtain a good approximation of the partition function provided $k_B T \ll P V_0$. Probability distribution (6) corresponds to a uniform distribution over the phase space coordinates of the system (\mathbf{s}, \mathbf{p}) , subject to the above constraints. [Recall that, in contrast, probability distribution (3) in Sec. II corresponds to a uniform distribution over the particle coordinates \mathbf{s} alone, subject to similar constraints.]

Since in TE-HMC we explicitly sample both coordinates and momenta, the nested sampling approximation to the

partition function becomes

$$\Delta(N, P, \beta) \approx \frac{\beta P}{N! h^{3N}} \sum_i (\Gamma_{i-1} - \Gamma_i) e^{-\beta H_i}, \quad (8)$$

where H_i is the total enthalpy of the i th nested sampling level, and Γ_i is the volume of phase space with total enthalpy less than or equal to H_i at pressure P .

In order to ensure the sampler spends approximately an equal amount of computer time exploring each degree of freedom, we set all the masses to be equal: $m_i = m \forall i$. We then recover the correct partition function (8) by multiplication:

$$\Delta(N, P, \beta) \approx \left(\prod_{i=1}^N \frac{m_i}{m} \right)^{\frac{3}{2}} \Delta_{\text{NS}}(N, P, \beta), \quad (9)$$

where Δ_{NS} is equal to the right-hand side of (8), calculated with equal particle masses. For equal particle masses, we find Γ_0 in Eq. (8) to be given by

$$\Gamma_0 = \frac{V_0^{N+1}}{N+1} \frac{2(2\pi m E_k^0)^{\frac{3N}{2}}}{3N \Gamma(\frac{3N}{2})}, \quad (10)$$

where $\Gamma(\frac{3N}{2})$ is the gamma function evaluated at $\frac{3N}{2}$.

In TE-HMC, the atomic coordinates \mathbf{r} and momenta \mathbf{p} are evolved according to the following Hamiltonian Monte Carlo sequence. The move begins with the initial Cartesian phase space coordinates $(\mathbf{r}^{(0)}, \mathbf{p}^{(0)})$, which are in the allowed region, $H < H^{\text{sup}}$:

(1) Randomize the momenta, either partially or completely, to pick new momenta satisfying $E_k(\mathbf{p}) < \min[E_k^0, H^{\text{sup}} - \hat{H}]$, as in Eq. (6). This momentum randomization takes us to the coordinates $(\mathbf{r}^{(0)}, \mathbf{p}^{(1)})$.

(2) Starting from $(\mathbf{r}^{(0)}, \mathbf{p}^{(1)})$, integrate Newton's equations of motion for the coordinates (\mathbf{r}, \mathbf{p}) over a fixed number of time steps. At the end of this trajectory the phase space coordinates are $(\mathbf{r}^{(1)}, \mathbf{p}^{(2)})$.

(3) Reverse the momenta $\mathbf{p}^{(3)} = -\mathbf{p}^{(2)}$; the trajectory and this reversal taken together, $(\mathbf{r}^{(0)}, \mathbf{p}^{(1)}) \rightarrow (\mathbf{r}^{(1)}, \mathbf{p}^{(3)})$, are a reversible MC proposal, which ensures that the move satisfies detailed balance.

(4) Calculate the new total enthalpy $H_{\text{trial}} = H(\mathbf{r}^{(1)}, V, \mathbf{h}_0, \mathbf{p}^{(3)})$. If $H_{\text{trial}} < H^{\text{sup}}$ and $E_k(\mathbf{p}^{(3)}) < E_k^0$ then accept the new coordinates $(\mathbf{r}^{(1)}, \mathbf{p}^{(3)})$; otherwise return

to the starting coordinates $(\mathbf{r}^{(0)}, \mathbf{p}^{(0)})$. The coordinates are now $(\mathbf{r}^*, \mathbf{p}^*)$.

(5) Reverse the momenta again. The final, resulting coordinates are $(\mathbf{r}^*, -\mathbf{p}^*)$.

A great advantage of the TE-HMC algorithm is that numerical integration of Newton's equations of motion approximately conserves the total enthalpy along the trajectory such that the value only fluctuates by a small amount. Consequently, the trial coordinates $(\mathbf{r}^{(1)}, \mathbf{p}^{(3)})$ nearly always satisfy $H_{\text{trial}} < H^{\text{sup}}$. Thus if partial momentum randomization is used, better preserving the direction of motion of the particles, one obtains excellent continuation between successive short TE-HMC trajectories. Pseudocode for the TE-HMC move introduced above is given in Appendix C. In particular, Algorithm 3 introduces the parameter γ , which controls the extent to which the direction of motion of the particles is randomized when using partial momentum randomization.

C. Wider application of NS with TE-HMC

The TE-HMC method can be used in applications of NS as a general method for Bayesian computation [2,3], by setting $m = 1$, equating $\hat{H}(\mathbf{r}) = -\ln \text{Prob}(\mathbf{r})$ where $\text{Prob}(\mathbf{r})$ is the Bayesian likelihood of the parameters \mathbf{r} , and specifying a suitably large value for the maximum kinetic energy, E_k^0 . The Bayesian evidence is approximated by

$$Z \approx \sum_i (\Gamma_{i-1} - \Gamma_i) e^{-H_i}$$

in which Γ_0 is given by

$$\Gamma_0 = \frac{2E_k^{0 \frac{d}{2}}}{3N \Gamma(\frac{d}{2})}$$

where $\Gamma(\frac{d}{2})$ is the gamma function evaluated at $\frac{d}{2}$, and d denotes the dimension of the parameter space: the phase space $\{\mathbf{r}, \mathbf{p}\}$ has $2d$ dimensions. Throughout this paragraph we have assumed that the prior over \mathbf{r} is uniform. For continuous problems it is always possible to use coordinates in which this is the case.

A total energy HMC algorithm, suitable for performing constant volume NS calculations, is obtained by replacing βP with 1 in Eq. (8) then setting the pressure $P = 0$ throughout the TE-HMC algorithm. Cell volume MC moves should not be performed, while cell stretch and cell shear moves may

TABLE I. Parameters for NS runs: pressure P , number of particles N , number of configurations K , number of configurations removed per iteration K_r , walk length (all-particle energy/force calls) L , step ratios (particle : cell volume : cell shear : cell stretch [: swap]), minimum temperature T_{min} , and number of NS iterations n_{iter} . Each particle step consisted of 8 N -particle energy/force evaluations: 8 all-particle sweeps for SP-MC, or a single 8 step trajectory for GMC and TE-HMC.

	P	N	K	K_r	L	step ratios	T_{min}	$n_{\text{iter}}/10^6$
mono LJ walk length	$3.162 \times 10^{-2} \epsilon / \sigma^3$	64	2304	1	80–2560	1:16:8:8	0.43 ϵ	2
binary LJ order-disorder	$3.162 \times 10^{-2} \epsilon / \sigma^3$	64	4608	2	1536	1:16:8:8:8	0.043 ϵ	3
Cu, Au	0.1 GPa	64	2304	1	768	1:16:8:8:8	600 K	2
Cu _x Au _{1-x} , $x = [0.25, 0.5, 0.75]$	0.1 GPa	64	4608	2	768	1:16:8:8:8	600 K	2
mW water	1.6 MPa	64	1920	1	3168	3:8:4:4	150 K	2
single-chain polymer	const. V	15	2304	1	5120	1:0:0:0	0.01 ϵ	1
multichain polymer cluster	const. V	8×15	4608	1	5120	1:0:0:0	0.3 ϵ	8
multichain polymer	$2.3 \times 10^{-3} \epsilon / \sigma^3$	8×15	4608	1	5120	1:4:4:4	1.2 ϵ	3

optionally be included or left off. Momenta are initialized as for total enthalpy HMC, with $E_k^0 = \frac{3N}{2}k_B T_0$ where T_0 is a high temperature which corresponds to the ideal gas. In total energy HMC one must replace Eq. (10) with

$$\Gamma_0 = V^N \frac{2(2\pi m E_k^0)^{\frac{3N}{2}}}{3N\Gamma(\frac{3N}{2})}. \quad (11)$$

This total energy HMC algorithm was used to perform the constant volume calculations reported in Sec. IV C 4.

IV. RESULTS

A. Parameters and implementation

Here we present tests of the performance of the constant-pressure NS sampling method with the different particle motion algorithms. The single-particle MC moves (SP-MC) are grouped into sweeps over the system moving each particle in random order. When the potential is separable, for example the LJ model in the examples below, the cost of an entire N -particle sweep is equal to a single full-system energy/force evaluation. When discussing walk lengths below we therefore consider an N -particle sweep equivalent to a single energy/force evaluation in other moves (GMC or TE-HMC for particle positions, as well as cell moves). For interactions of more general form that are not separable, the sweep would be N times slower than an energy/force evaluation.

Step sizes for all types of MCMC moves are automatically adjusted during the NS iteration process using pilot walks (which are not included in the NS configuration evolution) so as to reach acceptance rates of 0.5–0.95 for TE-HMC MD trajectories, and 0.25–0.75 for all other moves (cell steps, single-particle SP-MC steps, GMC trajectories). The essential NS parameters for all systems presented here are listed in Table I, and input files are provided in the Supplemental Material (SM) [13]. Each particle step consisted of 8 N -particle energy/force evaluations: 8 all-particle sweeps for SP-MC, or a single 8-step trajectory for GMC and TE-HMC. For multicomponent systems 8 swap steps were done in addition [1]. For TE-HMC partial randomization of the velocity direction was done as in Algorithm 3 with $\gamma = 0.3$, except for the polymer system which used $\gamma = 0.1$.

B. Efficiency of the SP-MC, GMC, and TE-HMC algorithms

As a basic test of the effectiveness of the three atomic motion algorithms, SP-MC, GMC, and TE-HMC, we apply the NS method to a periodic Lennard-Jones system with interactions truncated and shifted to zero at a cutoff distance of 3σ (see Appendix D 1). We choose a pressure above the triple point but below the critical point, where this system has two phase transitions: condensation from the gas to the liquid, and freezing from the liquid to the crystalline solid. Figure 1 compares the performance of the SP-MC, GMC, and TE-HMC algorithms for resolving the heat capacity peaks associated with these transitions. The cost of each GMC time step is similar to that in TE-HMC, and from Fig. 1 we see that the convergence of all three methods is similar for the condensation transition. However, TE-HMC is significantly more efficient than GMC for accurately resolving the freezing transition.

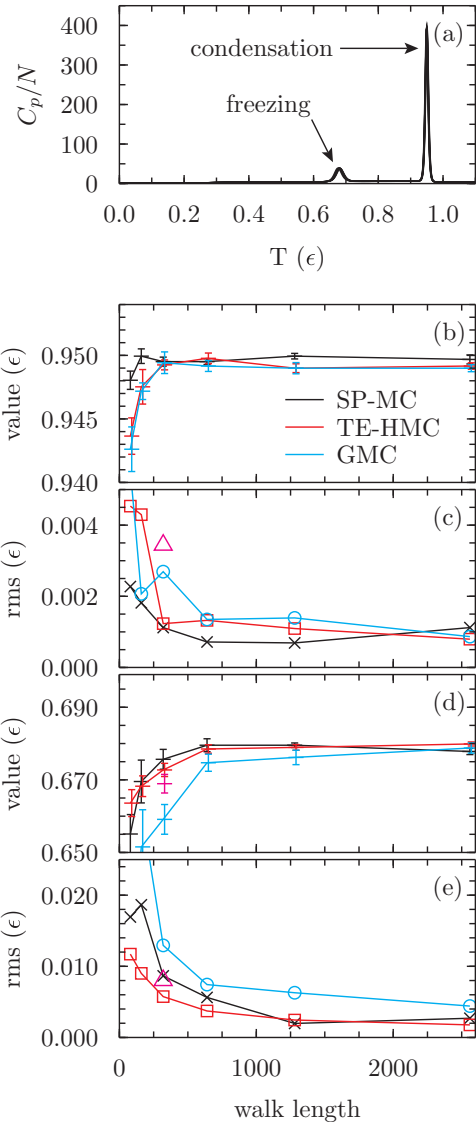


FIG. 1. Heat capacity curves $C_p(T)$, peak positions, and peak scatter calculated using NS for 64 LJ particles at pressure $P\sigma^3/\epsilon = 3.162 \times 10^{-2}$. Panel (a): Example $C_p(T)$ curve for 10 independent runs with SP-MC moves and $L = 1280$ energy evaluations. Panels (b)–(e): Convergence of the condensation [(b), (c)] and freezing [(d), (e)] transition peak temperatures as a function of walk length. For each transition, the upper panel [(b), (d)] shows the mean peak position (error bars indicate \pm one standard deviation from 10 independent runs), and the lower panel [(c), (e)] shows the root mean squared scatter in transition temperature values.

From Fig. 1 we can see that, for equivalent accuracy, the number of all-particle sweeps required by the SP-MC algorithm is similar to the number of time steps required by the TE-HMC and GMC algorithms. For separable potentials such as LJ, an all-particle sweep takes roughly the same time as the full energy/force evaluation used in a time step of GMC or TE-HMC. Therefore, for a nonseparable potential, GMC and TE-HMC are N times faster than the SP-MC algorithm.

The transition temperatures for the shorter, underconverged walks are systematically lower than the fully converged value. This underestimation is unsurprising: if too few steps are taken

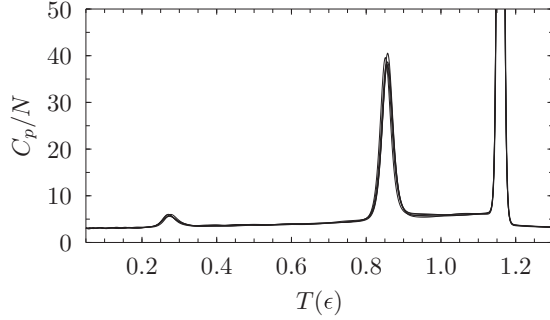


FIG. 2. Heat capacity as a function of temperature for a LJ binary alloy at equal composition for five independent NS runs. The three peaks correspond to condensation, freezing, and chemical ordering, from high T to low T .

at each iteration, then the sampler requires more iterations to find the low-temperature phase. Since subsequent NS iterations correspond to lower entropies and lower temperatures, finding the structure at a later iteration corresponds to an underestimated transition temperature. The root mean square (rms) scatter in peak positions shown in Fig. 1 does not go to zero for any of the methods, even with the longest walks used ($L = 2560$ energy/force evaluations): for infinite walk lengths, the accuracy is limited by the number of walkers, K , and the number removed at each iteration, K_r .

C. Example applications

In this section we demonstrate the utility of NS by applying it to study four diverse systems: the order-disorder transition of a binary LJ alloy, the eutectic of a copper-gold alloy, the density anomaly of water which forms open crystal structures, and the phase behavior of a bead-spring polymer model. In all four cases we use TE-HMC to explore the position degrees of freedom since it is most efficient for this range of system sizes, as discussed below in Sec. IV D. All simulations with the exception of the binary LJ were carried out with the LAMMPS package [9].

1. Order-disorder transition

In addition to the condensation, freezing, and martensitic transitions that have previously been simulated using NS [1], multicomponent solids also show transitions related to chemical ordering. Here we use NS to simulate the order-disorder transition of a model binary LJ alloy. The potential energy function used, which favors the mixing of atoms, is given in Appendix D 2.

Figure 2 shows the heat capacity curve of this system. The condensation, freezing, and order-disorder transitions can be seen as three separate peaks. Chemical ordering of the alloy can be observed in Figs. 3 and 4 which respectively show ensemble-averaged radial distribution functions (RDFs) and typical configurations of the alloy at temperatures corresponding to the ordered and disordered solids, as well as the liquid. Both the RDFs (Fig. 3) and the representative atomic configurations (Fig. 4) show that at $T = 0.95$, the system is a liquid, while at $T = 0.34$ the system is a chemically disordered close-packed crystal, and at $T = 0.17$ the crystal is chemically ordered.

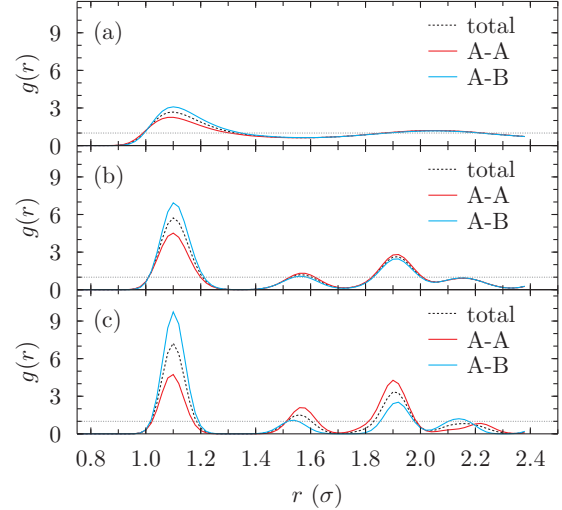


FIG. 3. Ensemble-averaged radial distribution functions from NS phase diagram of binary LJ at equal composition. Panel (a) is above the melting point, (b) is between the freezing and chemical ordering transitions, and (c) is below the chemical ordering transition. Liquid (a) shows a low first-neighbor peak and minimal other structure. Disordered solid (b) shows distinct peaks and some chemical ordering at first neighbors only. Ordered solid (c) shows stronger chemical ordering, at least out to third neighbors. These RDFs were calculated using a weighted sum of the RDFs of all configurations output by NS. The RDF for each configuration was weighted by its Boltzmann weight $(\Gamma_{i-1} - \Gamma_i)e^{-\beta H_i}$, as in the partition function Eq. (8).

2. Copper-gold eutectic

A eutectic, where the melting point of a multicomponent alloy is reduced at intermediate compositions due to entropic effects, is an important example of the interplay between energy and entropy affecting a phase transition. We used NS to compute the heat capacity of $\text{Cu}_x\text{Au}_{1-x}$ at a pressure of 0.1 GPa with interactions described by a simple Finnis-Sinclair-type embedded-atom model (FS-EAM) interparticle potential [14–16] (see Appendix D 3). In Fig. 5 we show the heat capacity $C_p(T)$ for a number of composition values x at a temperature range near the melting point, and in Fig. 6 we compare the resulting computed melting points to experimental results from Ref. [17].

We find that the FS-EAM potential gives a melting temperature $T_m \approx 1300$ K for pure Cu and $T_m \approx 1240$ K for pure Au. The computed melting temperature is lower at all intermediate

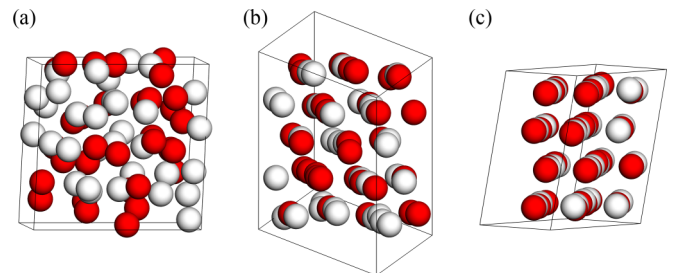


FIG. 4. Visualizations of typical configurations for the liquid phase [$T = 0.95$; (a)], chemically disordered solid [$T = 0.34$; (b)], and chemically ordered solid [$T = 0.17$; (c)].

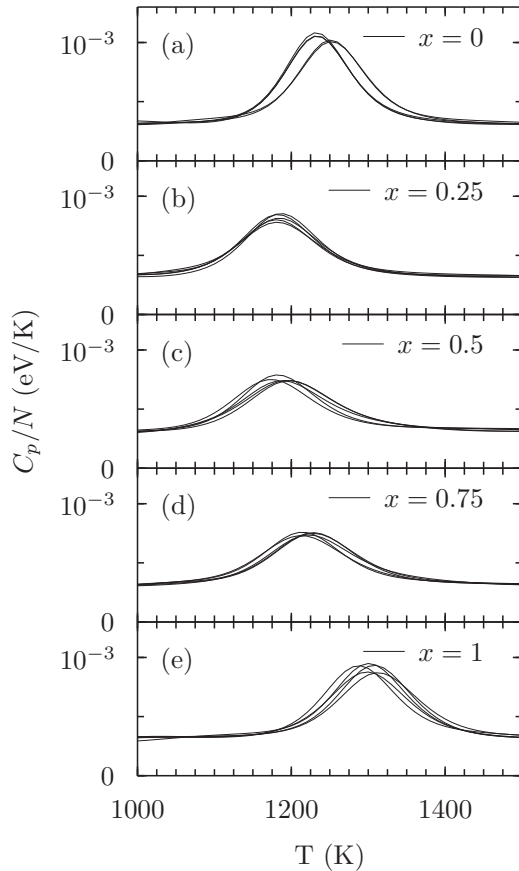


FIG. 5. Heat capacity as a function of temperature $C_p(T)$ of $\text{Cu}_x\text{Au}_{1-x}$ for a range of compositions, from pure Au $x = 0$ [panel (a)] to pure Cu $x = 1$ [panel (e)] in 25% intervals, showing variation of melting point peak. Each curve corresponds to an independent NS calculation.

compositions, with a minimum value in the range 1175–1190 K between 25% and 50% Cu; the computed melting temperatures at these two compositions are equal to within the error bars of the calculation. While this FS-EAM potential underestimates the experimental melting points of the two endpoints, more severely so for Au (by 8%), nested sampling shows that it reproduces the qualitative features of a eutectic.

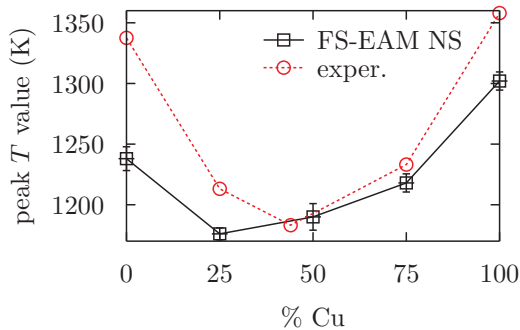


FIG. 6. Melting point as a function of composition in binary FS-EAM $\text{Cu}_x\text{Au}_{1-x}$ [14–16], calculated using the TE-HMC NS algorithm. Experimental values taken from [17]. Eutectic suppression of the melting point is observed at intermediate compositions.

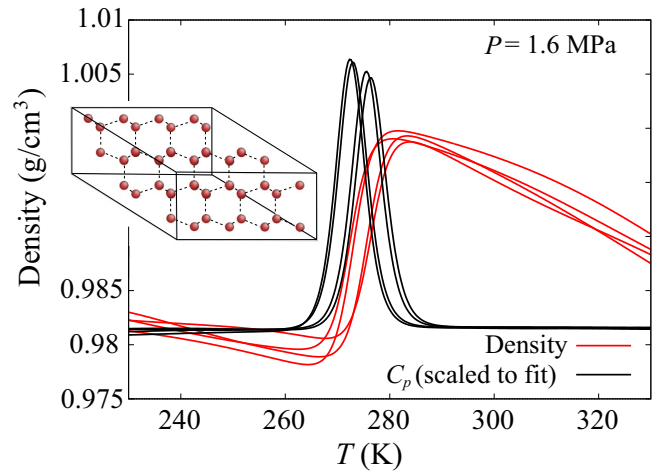


FIG. 7. Heat capacity C_p (in arbitrary units) and density curves for 64 mW water particles at a pressure of 1.6 MPa, calculated using the TE-HMC NS algorithm. The inset shows a visualization of the hexagonal ice structure found by NS. Dashed lines in the inset represent the hydrogen-bond network.

3. Density anomaly of water

The mW potential [18] is a coarse-grained model of water, designed to mimic its hydrogen-bonded structure through a nonbonding angular term, which biases the model towards tetrahedral coordination. Despite only having short-range interactions and a single particle representing a water molecule, it reproduces the energetics, anomalies, liquid, and hexagonal ice structure of water remarkably well. In order to demonstrate that NS is capable of finding not just close-packed but also open structures, we simulated water using the mW potential. The computed heat capacity and density curves from four calculations are shown in Fig. 7. As expected, the particles were observed to form a hexagonal ice structure, also shown in Fig. 7. By averaging the results from these calculations, we calculated the freezing temperature to be 274.3 ± 1.0 K, the density of ice to be 0.9792 ± 0.0005 g/cm³, and the density of the liquid at 298 K to be 0.9966 ± 0.0004 g/cm³. All these results are in excellent agreement with values previously calculated for the mW water model: 274.6 K, 0.978 g/cm³, and 0.997 g/cm³, respectively [18,19]. We found the maximum density of water to be 0.9992 ± 0.0002 g/cm³ at a temperature 8.1 ± 0.3 K above the freezing temperature.

4. Molecular solids

Molecular materials are another system where nested sampling can be used to efficiently sample the configuration space. Both single-molecule systems and multimolecule systems, such as aggregating proteins or polymer melts, are of interest. In previous studies Wang-Landau sampling was used to map the phase behavior of single polymer chains with different lengths [20,21] and bending stiffnesses [22,23]. Here we present results for a bead-spring polymer model (see Appendix D4) with a harmonic bond and cosine angle potential parameterized by stiffnesses k_b and k_a , respectively, and a nonbonded LJ interaction with energy ϵ . This model has been used to study crystallization in polymers [24], and is similar to the model used in a previous Wang-Landau study [20].

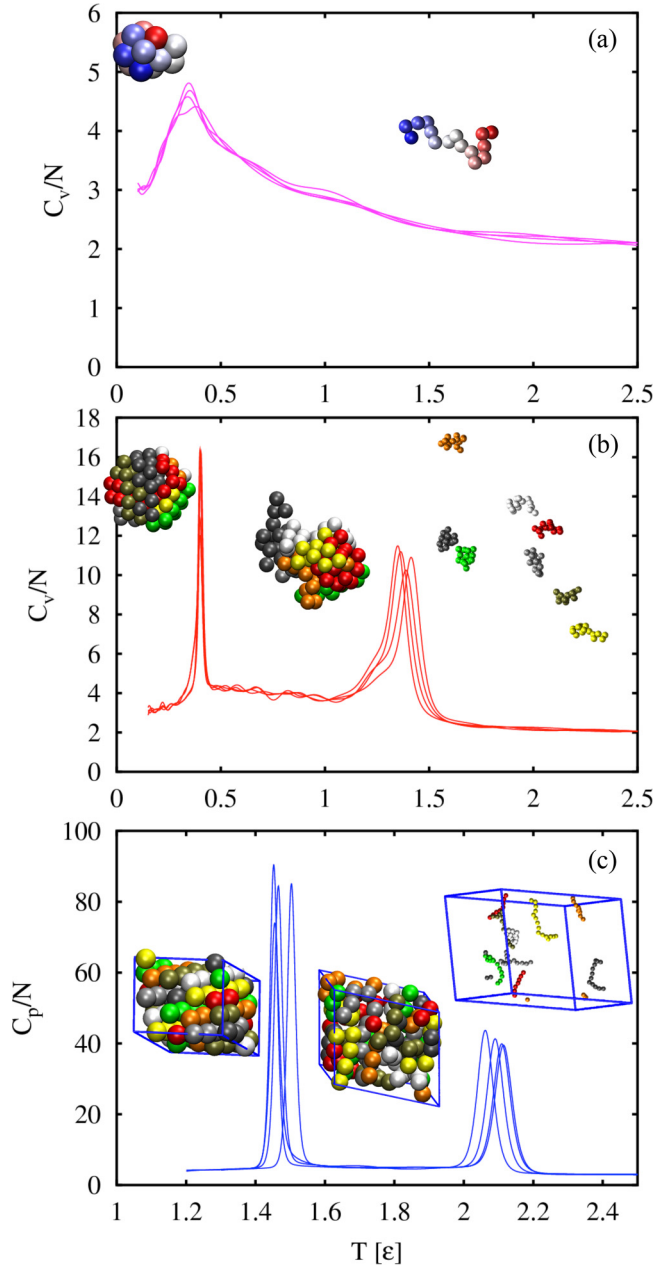


FIG. 8. Heat capacity per coarse-grained bead C_p/N or C_v/N curves for a single bead-spring polymer chain of 15 beads with monomer density $2.5 \times 10^{-5} \sigma^{-3} \approx (\frac{1}{40\sigma})^3$ [panel (a)], 8 15-bead chains in a periodic box with monomer density $2 \times 10^{-3} \sigma^{-3} \approx (\frac{1}{8\sigma})^3$ [panel (b)], and 8 15-bead chains in a periodic box with cell moves and $P = 2.3 \times 10^{-3} \epsilon/\sigma^3$ [panel (c)]. Both constant-volume systems use fully flexible ($k_a = 0$) models, while the constant-pressure system (bottom) has $k_a = 10\epsilon$. Snapshots show example polymer conformations corresponding to the different phases.

Figure 8 shows heat capacity curves calculated for three different systems: (i) a single, fully flexible ($k_a = 0$) bead-spring polymer chain of 15 beads in a constant-volume periodic box with monomer density $2.5 \times 10^{-5} \sigma^{-3} \approx (\frac{1}{40\sigma})^3$; (ii) 8 fully flexible 15-bead chains in a constant-volume periodic box with monomer density $2 \times 10^{-3} \sigma^{-3} \approx (\frac{1}{8\sigma})^3$; (iii) 8 15-bead chains with angle stiffness $k_a = 10\epsilon$ at fixed

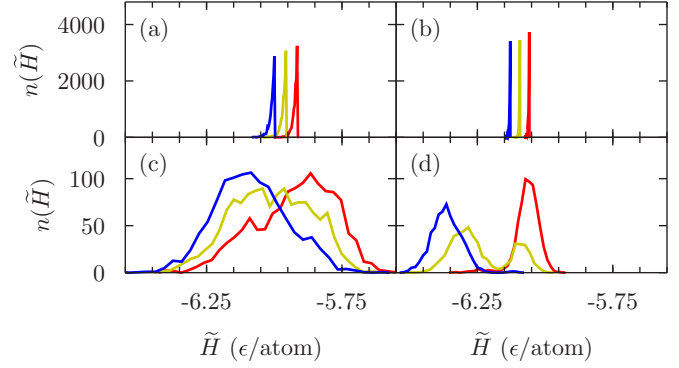


FIG. 9. Distribution of configurational enthalpy (excluding kinetic energy contribution for TE-HMC) for a 64-atom system [left column (a), (c)] and 256-atom system [right column (b), (d)] of monatomic LJ, for a range of NS iterations that spans the highest weight configurations for the freezing transition at $T \sim 0.65\epsilon$. Top row [(a), (b)] shows GMC results and bottom row [(c), (d)] shows TE-HMC. Colors indicating iteration, from earliest (highest energy) to latest (lowest energy) are red, yellow, and blue.

pressure $P = 2.3 \times 10^{-3} \epsilon/\sigma^3$, and flexible periodic boundary conditions. The low monomer densities of the constant-volume systems do not allow for a single chain to interact with itself through the periodic boundary at the temperatures of interest. Snapshots of configurations for each system type are also shown, illustrating the observed phase transitions.

The single chain shows a broad transition below $T = 0.5\epsilon$, from an extended state to a collapsed, ordered state, in agreement with previous results [20]. For short, unentangled polymer chains nested sampling leads to slightly lower relative error in the peak height when compared to previous results at approximately half the computational cost [20,23]. The constant-volume multichain system has two transitions: first, chain aggregation occurs at $T = 1.4\epsilon$, and second, at $T = 0.4\epsilon$ the monomers order, forming a solid cluster with high symmetry. This is reminiscent of the $N = 100$ single-chain transition observed previously [20]. The multichain periodic system has two transitions, the first at $T = 2.05\epsilon$ from a polymer gas to a melt, and the second at $T = 1.45\epsilon$ from a melt to a crystalline solid, in agreement with MD simulations of polymer crystal nucleation [24].

D. System size dependence of enthalpy distribution

As mentioned in Sec. III A, nested sampling using GMC creates a series of probability distributions (3) that correspond to uniform distributions in the Cartesian particle coordinates, \mathbf{r} , such that $\tilde{H}(\mathbf{r}) < \tilde{H}^{\text{sup}}$. In this case $\text{Prob}(\tilde{H})$ is proportional to the density of states for \tilde{H} , which is strictly unimodal. In contrast, TE-HMC works by performing nested sampling in total phase space, and samples from a series of probability distributions (6) that correspond to uniform distributions in the phase space coordinates (\mathbf{r}, \mathbf{p}) , with $H(\mathbf{r}, \mathbf{p}) < H^{\text{sup}}$. In TE-HMC, the marginal distribution for \mathbf{r} is *not* a uniform distribution, and for larger system sizes a bimodality is observed in the probability distribution for \tilde{H} at phase transitions.

Figure 9 compares the observed probability distributions for \tilde{H} in the region of the freezing transition (for the same pressure

as the system presented in Sec. IV B) with the TE-HMC and GMC algorithms, for simulations of 64 particles (as used in the earlier subsection) and also for 256 particles. Using TE-HMC for 64 particles $\text{Prob}(\tilde{H})$ is unimodal and broadens slightly at the freezing transition, but never becomes bimodal. For the larger 256-particle system, $\text{Prob}(\tilde{H})$ becomes bimodal, which can be clearly seen in the middle curve (yellow).

In order to obtain an accurate estimate of the integrated density of states, $\Gamma(H)$, TE-HMC must draw a sample from the uniform distribution in (\mathbf{r}, \mathbf{p}) , Eq. (6), at each iteration. NS approaches each transition from above, and if $\text{Prob}(\tilde{H})$ is bimodal, initially all K configurations will be in the mode at higher \tilde{H} . To draw a proper sample from $\text{Prob}(\tilde{H})$ at the phase transition, the MCMC walk must be long enough that the configuration can feasibly pass back and forth between the two modes, traversing the intermediate range of \tilde{H} , several times. For larger N , as $\text{Prob}(\tilde{H})$ gets smaller in the region between the modes, transitions between the two modes will become less frequent and much longer MCMC trajectories will be required at iterations close to the phase transition.

We observed in Sec. IV B that, for simulations of 64 particles, TE-HMC is significantly more efficient than GMC for accurately resolving the freezing transition. However, as a result of the bimodality in $\text{Prob}(\tilde{H})$, GMC may become more efficient than TE-HMC for larger system sizes. In the future, it would be desirable to develop algorithms which, like TE-HMC, use atomic forces at every step to expedite configuration space exploration, yet avoid this bimodality in $\text{Prob}(\tilde{H})$ at larger system sizes.

V. CONCLUSIONS

In this paper we have proposed efficient all-particle moves using interparticle forces and dynamics for constant-pressure nested sampling. The TE-HMC, GMC, and SP-MC algorithms reach the same accuracy using approximately the same number of full system energy evaluations (TE-HMC and GMC) or full SP-MC sweeps. For separable potentials, where a single-particle move can be computed in $1/N$ the cost of a full system energy evaluation, this makes the three methods equally efficient; for nonseparable potentials, where such efficient single-atom moves are not possible, the TE-HMC and GMC algorithms are N times faster.

The TE-HMC algorithm uses constant-energy molecular dynamics, implemented in many software packages, but requires extending the NS method to sample positions and momenta, and leads to increasingly bimodal configurational enthalpy distributions as the system size increases. This bimodality is likely to make equilibration and sampling difficult for sufficiently large systems, although this has not been a practical problem for the 64–120 particle systems we have considered here. The GMC algorithm, although somewhat less efficient in this size range, maintains the unimodal configurational enthalpy distribution of the previous Gibbs-sampling-based approach, and is therefore not expected to suffer from a breakdown in ergodicity for larger systems.

We have implemented the constant-pressure NS method using these algorithms in the PYMATNEST software [8], which includes a parallel algorithm and a link to the LAMMPS package which itself has many interparticle potentials available. Using

this implementation we have shown that the constant-pressure NS method with these algorithms can be used to simulate a wide range of systems with different interaction potentials and types of phase transitions: order-disorder transitions in binary LJ, eutectic composition dependence of the melting point in Cu-Au, freezing of water which has a density anomaly and an open crystal structure, and condensation and solidification of a bead-spring polymer model.

ACKNOWLEDGMENTS

R.J.N.B. acknowledges support from EPSRC Grant No. EP/J017639/1. G.C. acknowledges support from EPSRC under Grants No. EP/P022596/1 and No. EP/J010847/1. L.B.P. acknowledges support from the Royal Society. The work of N.B. was supported by the Office of Naval Research through the US Naval Research Laboratory's 6.1 base program. K.M.S. was supported by the National Research Council Associateship Program at the US Naval Research Laboratory. N.B. and K.M.S. acknowledge computational resources from the DOD High Performance Computing Modernization Program Office (HPCMPO) at the ARL and AFRL DSRCs.

APPENDIX A: NS STOPPING CRITERIA: TEMPERATURE ESTIMATES

The simplest criteria for stopping the NS iteration are to use a fixed number of iterations (i.e., fixed reduction in entropy) or a fixed potential energy minimum (e.g., close to the ground state). A more physically appealing criterion takes advantage of the approximate correspondence between the downward scan in enthalpy and decreasing temperature. The estimate we use to terminate the outermost NS iteration loop is based on the expressions for thermodynamic quantities, such as the partition function (or enthalpy or heat capacity, which are its derivatives). We find that the range of iterations that contribute with significant weight to the ensemble average at each temperature is sharply peaked. When the contribution of the current iteration to the partition function at a specified temperature T_{\min} is a factor of e^{10} lower than the maximum contribution of any previous iteration, we assume that no later iteration will contribute significantly, and therefore consider the calculation to be converged for all $T \geq T_{\min}$. We use this stopping criterion in all simulations reported here. Note that a monotonic relationship between iteration and temperature is not always satisfied; near phase transitions the dependence is more complicated, and setting T_{\min} too near a phase transition will lead to unpredictable behavior.

The convergence criterion described above is efficient enough to evaluate at each iteration for a single choice of T_{\min} , but too computationally expensive to use as an estimate of the “current temperature” during NS iterations, because it would need to be evaluated for many values of T to find the lowest. We therefore use an independent estimate of the temperature to monitor the progress of the NS iterations. This estimate is based on the rate of decrease of \tilde{H}^{sup} (or H^{sup}) as a function of iteration number. The iteration number i is linearly related to the logarithm of configuration space volume (i.e., microcanonical entropy S), and that rate of decrease is therefore related to $\partial \tilde{H} / \partial S$. The current temperature during

the NS simulation can be estimated from the finite-difference expression

$$T \approx \left(k_B \frac{D \ln \alpha}{\tilde{H}^{\text{sup}}(i - D) - \tilde{H}^{\text{sup}}(i)} \right)^{-1}, \quad (\text{A1})$$

where D is an interval over which the finite difference is taken (1000 iterations here), and k_B is Boltzmann's constant. We use this expression to monitor the progress of the NS iterations, but not to terminate.

APPENDIX B: PARALLELIZATION

The PYMATNEST software [8], in which these algorithms are implemented, combines two separate forms of parallelization which were previously reported separately. In Ref. [1], during step 2 of the NS algorithm, rather than decorrelating a single cloned configuration alone using a MCMC walk comprising L energy evaluations, the authors evolve n_p configurations (including the cloned configuration) in parallel using n_p processes through L/n_p MC moves each. Each configuration is evolved for an average of n_p iterations before being recorded and removed. Thus the user specifies the *average* number of energy evaluations used to decorrelate a cloned configuration from its starting coordinates. In Refs. [25,26], on the other hand, the authors evolve each cloned configuration for exactly L steps, but they parallelize over n_p processes by removing $K_r = n_p > 1$ configurations at each NS iteration, resulting in K_r cloned configurations that can be walked in parallel.

In PYMATNEST, we combine the two formulations by allowing for $K_r > 1$ and evolving the K_r cloned configuration in parallel, but also for the number of parallel tasks $n_p > K_r$, reducing the walk length required at each iteration. To optimize load balance, each of the K_r cloned configurations that must be evolved is assigned to a different parallel task, and all remaining parallel tasks (which would otherwise be idle if $n_p > K_r$) walk $K_e = n_p - K_r$ additional randomly chosen configurations.

From the probabilities for a configuration to be removed or walked at each NS iteration it is possible to calculate the distribution of the number of walks each configuration has experienced, and from that the mean number of times a configuration will be walked before it is removed ($\langle n_{\text{walks}} \rangle$). The general expression for the length of the walk that must be done at each iteration to achieve an expected total walk length $\langle L \rangle$, for arbitrary K, K_r , and K_e , is

$$L' = \langle L \rangle / \langle n_{\text{walks}} \rangle = \langle L \rangle \frac{K_r}{K_r + K_e} = \langle L \rangle \frac{K_r}{n_p}. \quad (\text{B1})$$

For each MCMC walk in PYMATNEST, different move types are randomly chosen from the list of possible moves with predetermined ratios until at least L' energy evaluations have been performed.

To maintain load balance the shortest walk must be a few times longer than the longest possible single step, for example, a single SP-MC sweep or an MD/GMC trajectory. Therefore the maximum parallelization, n_p , that can be achieved depends only on the total walk length, L , and number of configurations removed at each iteration, K_r , and not on the number of configurations, K . For typical runs we show here, $K_r = 1, L \approx 500$ –1000, and the length of each MD trajectory is 8. To keep

reasonable parallel efficiency we find that L' must be larger than about 20, so the maximum $n_p \approx 25$ –50.

1. Qualitative behavior of parallelized NS

The computational cost and accuracy of NS depend on these parameters in a complex way. The total computational work is proportional to K and $\langle L \rangle$, and is independent of K_r and n_p . If K is increased at fixed K_r , the fraction of configuration space that remains after each iteration $\alpha = (K + 1 - K_r)/(K + 1)$ comes closer to 1.0, and the number of NS iterations required increases approximately linearly with K . If instead K and K_r are increased proportionately, α and therefore the number of iterations remain roughly constant, but the work at each iteration (to walk K_r cloned configurations) increases proportionately. It is not clear *a priori* how the necessary value of $\langle L \rangle$ changes with K : there is some evidence that once K is large enough, increasing it further reduces the distance each cloned configuration must be walked to decorrelate it sufficiently, but this relationship requires further investigation.

The accuracy of the configuration space volume estimates computed by NS also depends on K and K_r . The value of α determines the resolution in configuration space volume, but larger values of K (at constant α) reduce the *noise* in the estimate (for the same reason that the 500th sample out of 999 is a less noisy estimator of the median than the 2nd sample out of 3). The value of $\langle L \rangle$ also affects the error, because insufficiently walked configurations have a correlation to the configuration they were cloned from, which leads to a deviation from the uniform distribution.

Since the useful parallelism is limited by the minimum value of L' , which is clearly independent of K , only increasing $\langle L \rangle$ or K_r can increase it. The former is useful only up to the point where the configurations are sufficiently decorrelated, as our convergence plots in Sec. IV B show. Increasing the latter at constant K decreases the resolution in configuration space volume (by decreasing α), and therefore leads to increased error. Increasing K_r while also increasing K proportionately maintains the resolution and actually reduces the noise in the configuration space volumes, but also increases the computational work, but not necessarily the time to solution if n_p can also be increased proportionately.

2. Quantitative behavior of parallelized NS

In this section we limit the discussion to the original constant-pressure, flexible periodic boundary conditions nested sampling algorithm as implemented by SP-MC and GMC. The same discussion can be extended exactly to TE-HMC by a change of symbols: H for \tilde{H} and Γ for χ (see Sec. III), so long as one takes care not to confuse the phase space volume Γ and the gamma function in Appendix B 2 a. In the next two subsections we separately discuss the effect of the two approaches to parallelization: first varying K_r while assuming our MCMC draws perfect samples from the probability distribution (3) [or (6)]; second varying K_e at fixed $\langle L \rangle$ and K_r .

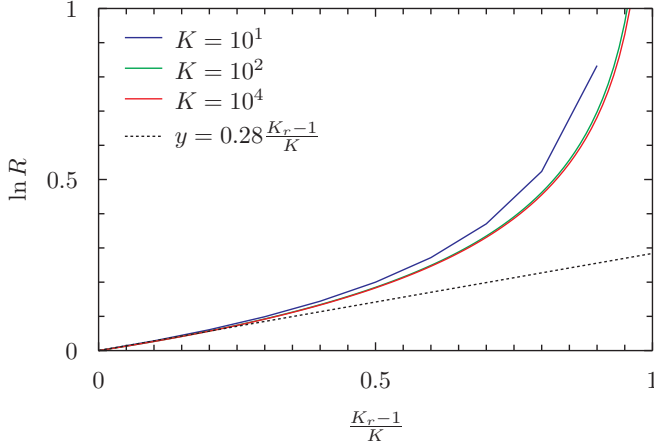


FIG. 10. Ratio of the rates, R , at which uncertainty in $\ln \chi$ accumulates for runs parallelized by removing and walking $K_r > 1$ as compared to serial run with $K_r = 1$, as a function of scaled number of configurations removed $\frac{K_r - 1}{K}$, for several values of K . Dashed line indicates linear trend for $K_r - 1 \ll K$.

a. $K_r > 1$ and $K_e = 0$

In this subsection, we assume that our MCMC walk yields perfect samples from the distribution (3) [or (6)]. In step 1 of the NS algorithm (see Sec. II) \tilde{H}^{sup} is updated to the lowest of the K_r highest enthalpies in our sample set, and the configuration space volume contained by the updated \tilde{H}^{sup} is $\chi_i \approx \chi_0 [(K - K_r + 1)/(K + 1)]^i$, where i is the NS iteration number. For $K_r > 1$ it is also possible to give analytic estimates of the configuration space volumes contained by the $K_r - 1$ higher enthalpy values between $\tilde{H}_{i-1}^{\text{sup}}$ and \tilde{H}_i^{sup} [26]. Thus one may consider the configurational entropy contained at fractional numbers of NS enthalpy levels.

After a number of enthalpy levels

$$n_\Delta = \left(\sum_{i=K-K_r+1}^K \frac{1}{i} \right)^{-1} \quad (\text{B2})$$

the expectation of the logarithm of the configuration space enclosed by \tilde{H} decreases by 1:

$$\langle \ln \chi_i - \ln \chi_{i+n_\Delta} \rangle = -1. \quad (\text{B3})$$

If we assume that it is possible to draw perfect random samples from (3) [or (6)], then it can be shown that, after the same number of enthalpy levels n_Δ , the variance of $\Delta \ln \chi = \ln \chi_i - \ln \chi_{i+n_\Delta}$ is given by

$$\text{Var}(\Delta \ln \chi) = \frac{d^{(K-K_r+1)} \Gamma(z)}{dz^{(K-K_r+1)}} \Big|_{z=1} - \frac{d^{(K+1)} \Gamma(z)}{dz^{(K+1)}} \Big|_{z=1}, \quad (\text{B4})$$

where $\Gamma(z)$ is the gamma function. The standard deviation, $[\text{Var}(\Delta \ln \chi)]^{\frac{1}{2}}$, represents the rate at which uncertainty in $\ln \chi$ accumulates during a nested sampling calculation. For a serial calculation ($K_r = 1$ and $K_e = 0$), $[\text{Var}(\Delta \ln \chi)]^{\frac{1}{2}} = \frac{1}{\sqrt{K}}$.

Figure 10 shows how the ratio of $[\text{Var}(\Delta \ln \chi)]^{\frac{1}{2}}$ for parallel and serial NS, $R = [\text{Var}(\Delta \ln \chi)]^{\frac{1}{2}} / \frac{1}{\sqrt{K}}$, depends on $\frac{K_r}{K}$. R represents the relative rate at which uncertainty in $\ln \chi$ accumulates during parallel and serial calculations. One can see that $\ln R$ converges for $K > 10^2$, and for $\frac{K_r}{K} \lesssim 0.25$, $[\text{Var}(\Delta \ln \chi)]^{\frac{1}{2}} \sim \exp(0.28 \frac{K_r - 1}{K}) \frac{1}{\sqrt{K}}$. For larger $\frac{K_r - 1}{K}$, R increases more rapidly.

b. $K_e > 0$ and $K_r = 1$

Parallelizing with $K_r = 1$ by using additional tasks to walk $K_e > 0$ extra configurations (in addition to those that were cloned) changes the walk length L from a deterministic parameter of the NS method to a stochastic one. It is possible to derive the variance of the walk length

$$\text{Var}(L) = \langle L \rangle^2 \frac{K_e}{K_e + K_r}. \quad (\text{B5})$$

While unlike the case of $K_r > 1$ there are no analytic results for the error due to the variability of L , several observations can be made. One is that the square root of the variance (except for the serial case of $K_e = 0$) is almost as large as L itself. Another is that the scaling of $\text{Var}(L)$ with the number of extra parallel tasks K_e is polynomial, unlike the exponential scaling of the error R with extra tasks for $K_r > 1$. It is unclear, however, how this variability in walk length will affect the error in the results, although the empirical observation is that this effect does not appear to be strong. Nevertheless, the serial case of $K_r = 1$ and $K_e = 0$ results in the lowest uncertainty in estimates of configuration or phase space volumes.

Algorithm 1. Total enthalpy Hamiltonian Monte Carlo step for nested sampling. Repeated application converges to uniform sampling of phase space in the region $H < H^{\text{sup}}, E_k(\mathbf{p}) < E_k^0$.

```

subroutine te_hmc(s,p,V,h0,M,dt)
! Nested sampling TE-HMC step

E_k^max = min[E_k^0, H^sup - H(s,V,h0)]

if(complete_momentum_rand):
    p = complete_rand_p(E_k^max)
else:
    p = partial_rand_p(E_k^max)

! NVE MD for M steps with time step dt
! Starts from (s,p), ends at (s',p')
(s',p') = MD_traj(s,p,V,h0,M,dt)

p' ← -p' ! reverse momenta: ensures
! detailed balance

if (H(s',V,h0;p') < H^sup AND E_k(p') < E_k^0):
    (s,p) ← (s',p') ! Accept proposal (s',p')

p ← -p ! reverse momenta

return (s,p)
end subroutine
    
```

APPENDIX C: TE-HMC MONTE CARLO STEP

The TE-HMC Monte Carlo step is given in Algorithm 1. This algorithm makes use of subroutines for complete and partial momentum randomization given in Algorithms 2 and 3, respectively.

We set all particle masses to be equal, as described in Sec. III B, which helps ensure the sampler spends an approximately equal amount of computer time exploring each degree of freedom.

Algorithm 2. Full momentum randomization from the uniform distribution inside the hypersphere $E_k(\mathbf{p}) < E_k^{\max}$. We set all particle masses to be equal, as described in Sec. III B. `uranf()` is a random number uniform in $[0,1]$. `granf()` is a Gaussian distributed random number, `Normal(0,1)`.

```

subroutine complete_rand_p( $E_k^{\max}, m$ )
  ! Random vector  $\mathbf{p}$  with  $E_k(\mathbf{p}) < E_k^{\max}$ .

  ! Generate random unit vector  $\hat{\mathbf{p}}$ 
   $\hat{\mathbf{p}}_i = \text{granf}() : i = 1, 3N$ 
   $\hat{\mathbf{p}} = \hat{\mathbf{p}}/|\hat{\mathbf{p}}|$ 

  ! Choose random  $|\mathbf{p}|$  with  $E_k(\mathbf{p}) < E_k^{\max}$ 
   $a = \text{uranf}()^{**}(1/(3N))$ 
   $\mathbf{p} = a(2mE_k^{\max})^{\frac{1}{2}}\hat{\mathbf{p}}$ 

  return  $\mathbf{p}$ 
end subroutine

```

Algorithm 3. Partial momentum randomization for TE-HMC nested sampling. Converges to the uniform distribution inside the hypersphere $E_k(\mathbf{p}) < E_k^{\max}$. We set all particle masses to be equal, as described in Sec. III B. `uranf(a,b)` is a random number uniform in $[a,b]$. Note, for odd numbers of atoms N , we do not rotate \mathbf{p} component `rand_indices(3N)`. However, since `rand_indices(3N)` is chosen at random, the subroutine `partial_rand_p` satisfies detailed balance.

```

subroutine partial_rand_p( $\mathbf{p}$ )
  ! Partial randomization of momentum  $\mathbf{p}$ .

  ! Choose random  $|\mathbf{p}|$  with  $E_k(\mathbf{p}) < E_k^{\max}$ 
   $a = \text{uranf}()^{**}(1/(3N))$ 
   $\mathbf{p} = a(2mE_k^{\max})^{\frac{1}{2}}\mathbf{p}/|\mathbf{p}|$ 

  ! indices  $\{1,2,\dots,3N\}$  in a random order
  rand_indices = random_order(1,3N)

  do  $i = 1, \text{floor}(3N/2)$  ! loop over pairs
    ! pick random angle from  $[-\gamma,\gamma]$ 
     $\theta = \text{urand}(-\gamma,\gamma)$ 

    ! pair of components of  $\mathbf{p}$  vector
     $j = \text{rand\_indices}(2i - 1)$ 
     $k = \text{rand\_indices}(2,i)$ 

    ! 2D rotation of  $\mathbf{p}$  components  $j,k$ 
     $u = \cos\theta \mathbf{p}_j + \sin\theta \mathbf{p}_k$ 
     $v = -\sin\theta \mathbf{p}_j + \cos\theta \mathbf{p}_k$ 
     $\mathbf{p}_j = u$ 
     $\mathbf{p}_k = v$ 
  end do

  return  $\mathbf{p}$ 
end subroutine

```

APPENDIX D: POTENTIAL ENERGY FUNCTIONS**1. Lennard-Jones potential**

The Lennard-Jones potential used in the paper is the “truncated and shifted” potential, given by

$$U(\mathbf{r}) = \begin{cases} 4\epsilon \sum_{i<j} \left[\left(\frac{\sigma}{r_{ij}}\right)^{12} - \left(\frac{\sigma}{r_{ij}}\right)^6 - c \right], & r_{ij} < r_c, \\ 0, & r_{ij} \geq r_c, \end{cases}$$

$$c = \left(\frac{\sigma}{r_c}\right)^{12} - \left(\frac{\sigma}{r_c}\right)^6. \quad (\text{D1})$$

Here, r_{ij} is the radial distance between particles i and j , while two atoms at dynamical equilibrium have a combined energy $-\epsilon$ and are separated by a distance $2^{\frac{1}{6}}\sigma$. This potential goes continuously to zero at a radius r_c . Calculations were performed using $\epsilon = 1, \sigma = 1, r_c = 3$.

2. Binary Lennard-Jones alloy potential

The potential used to simulate the binary Lennard-Jones alloy is given by

$$U(\mathbf{r}) = \begin{cases} 4 \sum_{i<j} \epsilon_{ij} \left[\left(\frac{\sigma}{r_{ij}}\right)^{12} - \left(\frac{\sigma}{r_{ij}}\right)^6 \right], & r_{ij} < r_c, \\ 0, & r_{ij} \geq r_c. \end{cases} \quad (\text{D2})$$

Calculations were performed using $\epsilon_{AA} = 1, \epsilon_{AB} = 1.5, \epsilon_{BB} = 1.5, \sigma = 1, r_c = 3$. In this potential, all atoms have equal atomic radii, but interactions between different atomic species (A and B) are 1.5 times stronger than A-A interactions or B-B interactions.

3. CuAu EAM

We used a Finnis-Sinclair-type embedded-atom model (EAM) for the $\text{Cu}_x\text{Au}_{1-x}$ binary alloy. The potential parameters are from the method in Ref. [14], based on the pure element parameters of Ref. [16], with interspecies parameters that are optimized to fit the formation energies of a few crystal structures at selected compositions of the binary alloy. The full parameter set for LAMMPS [9] is available for download from Ref. [15].

4. Bead-spring polymer models

The bead-spring models used to study long molecule chains are based on those used by Nguyen *et al.* to study polymer crystallization [24]. The energy of a bond between two monomers along the backbone of a polymer chain

$$U_b(\ell) = \frac{k_b}{2}(\ell - a)^2 \quad (\text{D3})$$

is harmonic in the distance from the characteristic distance a . The bond stiffness $k_b = 600\epsilon/\sigma^2$. The energy of an angle θ formed by three consecutive monomers along the polymer backbone is given by a cosine potential,

$$U_a(\theta) = k_a[1 - \cos(\theta)], \quad (\text{D4})$$

where the angular stiffness k_a penalizes angular deviations away from a straight backbone $\theta = 180^\circ$ and is set either to $k_a = 0$ for a fully flexible chain or to $k_a = 10\epsilon$. The

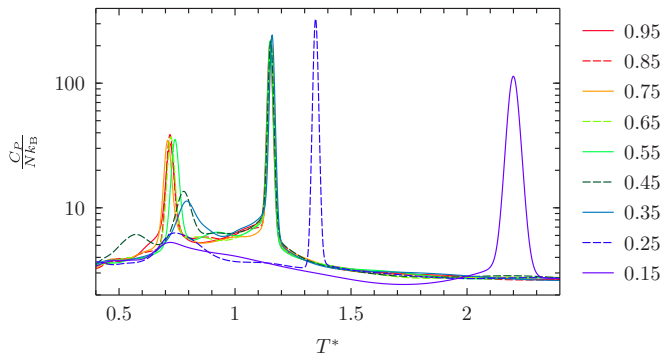


FIG. 11. Convergence of the heat capacity $C_p(T)$ with respect to minimum cell depth, d_0 , for a periodic system of 64 Lennard-Jones particles at pressure $\log_{10} P^* = -1.194$. The peak at high temperature corresponds to condensation, while the peak at lower temperature corresponds to freezing. The legend on the right shows the value of d_0 used in each calculation.

nonbonded interaction between two monomers a distance r from one another is given by Eq. (D1). The monomer diameter $\sigma = 2^{-1/6}a$, so that the bond distance and diameter are commensurate. The cutoff is set to $r_c = 3\sigma$. As the name suggests, nonbonded interactions apply only to monomers that are not bonded together.

APPENDIX E: MINIMUM CELL DEPTH d_0

Figure 11 shows the heat capacity of a periodic system of 64 Lennard-Jones particles at fixed pressure. This calculation used a potential similar to that given in Appendix D 1. In particular, we set the radial cutoff $r_c = 3\sigma$ and $c = 0$ in Eq. (D1). We also incorporated the standard long-range

correction to the energies to account for interactions beyond the cutoff [27]. Each curve corresponds to a single NS simulation, performed using SP-MC nested sampling, with $K = 640$, $K_r = 1$, $L = 2824$, and MC steps in the ratio (1 64-particle SP-MC sweep : 10 cell volume : 1 cell shear : 1 cell stretch). In these calculations each 64-particle SP-MC sweep was broken up into 64 individual single atom SP-MC moves, interspersed between cell moves. In each calculation we constrained the cell depth to be greater than some minimum value, d_0 [see Eq. (3)]. A clear transition to a quasi-2D system is observed when reducing d_0 . The location of the condensation transition is independent of d_0 for $d_0 \geq 0.35$, and the location of the freezing transition for $d_0 \geq 0.65$.

At low values of d_0 the simulation cell becomes very thin in at least one dimension and the system's behavior is dominated by unphysical correlations introduced by the periodic boundary conditions. The effect of unphysical correlations is reduced at lower densities, and also by increasing d_0 which constrains the simulation cell to more cubelike cell shapes. A larger value of d_0 is thus required at higher densities to sufficiently reduce the unphysical correlations. At the same time, setting d_0 too close to 1 excludes crystal structures that require a noncubic simulation cell. Quoting from [1], "The window of independence from d_0 grows wider as the number of particles is increased. For larger numbers of atoms, there are more ways to arrange those atoms into a given crystal structure, including in simulation cells that are closer to a cube. Similarly, unphysical correlations are introduced when the *absolute* number of atoms between faces of the cell becomes too small, and therefore larger simulations can tolerate 'thinner' simulation cells \mathbf{h}_0 ." It is clear from Fig. 11 that by imposing a suitable minimum cell height we can remove the unphysical behavior from the fully flexible cell formulation.

- [1] R. J. N. Baldock, L. B. Pártay, A. P. Bartók, M. C. Payne, and G. Csányi, *Phys. Rev. B* **93**, 174108 (2016).
- [2] J. Skilling, *AIP Conf. Proc.* **735**, 395 (2004).
- [3] J. Skilling, *J. Bayesian Anal.* **1**, 833 (2006).
- [4] D. Frenkel and B. Smit, *Understanding Molecular Simulations* (Academic Press, London, 2002), Chap. III, pp. 44–45.
- [5] S. Duane, A. Kennedy, B. J. Pendleton, and D. Roweth, *Phys. Lett. B* **195**, 216 (1987).
- [6] J. Skilling, *AIP Conf. Proc.* **1443**, 145 (2012).
- [7] M. Betancourt, *AIP Conf. Proc.* **1305**, 165 (2011).
- [8] N. Bernstein, R. J. N. Baldock, L. B. Pártay, J. R. Kermode, T. D. Daff, A. P. Bartók, and G. Csányi, PYMATNEST, <https://github.com/libAtoms/pymatnest>
- [9] S. Plimpton, *J. Comput. Phys.* **117**, 1 (1995).
- [10] L. B. Pártay, A. P. Bartók, and G. Csányi, *J. Phys. Chem. B* **114**, 10502 (2010).
- [11] R. L. Smith, *Oper. Res.* **32**, 1296 (1984).
- [12] L. Lovasz, *Math. Program. Ser. A* **86**, 443 (1999).
- [13] See Supplemental Material at <http://link.aps.org/supplemental/10.1103/PhysRevE.96.043311> for PYMATNEST input files sufficient to reproduce many of the calculations presented in this paper.
- [14] L. Ward, A. Agrawal, K. M. Flores, and W. Windl, [arXiv:1209.0619](https://arxiv.org/abs/1209.0619).
- [15] See <https://atomistics.osu.edu/eam-potential-generator/index.php>.
- [16] X. W. Zhou, R. A. Johnson, and H. N. G. Wadley, *Phys. Rev. B* **69**, 144113 (2004).
- [17] H. Okamoto, D. J. Chakrabarti, D. E. Laughlin, and T. B. Massalski, *Bull. Alloy Phase Diag.* **8**, 454 (1987).
- [18] V. Molinero and E. B. Moore, *J. Phys. Chem. B* **113**, 4008 (2009).
- [19] D. T. Limmer and D. Chandler, *J. Chem. Phys.* **135**, 134503 (2011).
- [20] D. T. Seaton, T. Wüst, and D. P. Landau, *Phys. Rev. E* **81**, 011802 (2010).
- [21] M. P. Taylor, W. Paul, and K. Binder, *J. Chem. Phys.* **131**, 114907 (2009).
- [22] D. T. Seaton, S. Schnabel, D. P. Landau, and M. Bachmann, *Phys. Rev. Lett.* **110**, 028103 (2013).
- [23] D. T. Seaton, Ph.D. thesis, University of Georgia, 2010.
- [24] H. T. Nguyen, T. B. Smith, R. S. Hoy, and N. C. Karayiannis, *J. Chem. Phys.* **143**, 144901 (2015).
- [25] N. S. Burkoff, C. Várnai, S. A. Wells, and D. L. Wild, *Biophys. J.* **102**, 878 (2012).
- [26] S. Martiniani, J. D. Stevenson, D. J. Wales, and D. Frenkel, *Phys. Rev. X* **4**, 031034 (2014).
- [27] D. Frenkel and B. Smit, *Understanding Molecular Simulations* (Academic Press, London, 2002), Chap. III, pp. 35–37.