

Optimal parallelisation strategies for flat histogram Monte Carlo sampling

Hubert J. Naguszewski,^{1,*} Christopher D. Woodgate,^{2,†} and David Quigley^{1,‡}

¹*Department of Physics, University of Warwick, Coventry, CV4 7AL, United Kingdom*

²*H.H. Wills Physics Laboratory, University of Bristol, Royal Fort, Bristol, BS8 1TL, United Kingdom*

(Dated: October 13, 2025)

Flat histogram methods, such as Wang–Landau sampling, provide a means for high-throughput calculation of phase diagrams of atomistic/lattice model systems. Many parallelisation schemes with varying degrees of complexity have been proposed to accelerate such sampling simulations. In this study, several widely used schemes are benchmarked—both in isolation and in combination—to establish best practice. The schemes studied include energy domain decomposition with both static sizing of energy sub-domains, as well as a dynamic sub-domain sizing scheme which we propose. We also assess the benefits both of replica exchange and of including multiple random walkers per sub-domain, to determine which factors have the largest impact on parallel efficiency. Additionally, the influence of energy sub-domain overlap regions is discussed. As an illustrative test case, we implement and apply the aforementioned strategies to a lattice-based model describing the internal energies of the AlTiCrMo refractory high-entropy superalloy, which is understood to crystallographically order into a B2 (CsCl) structure with decreasing temperature. We find that—while all of the proposed strategies confer a non-negligible speedup—parallelisation across energy domains which are non-uniform in size offers the most appreciable performance improvements. This work offers concrete recommendations for which parallelisation strategies should be prioritised to optimally accelerate flat-histogram Monte Carlo simulations.

Keywords: Parallelism, Monte Carlo methods, Wang–Landau sampling, Alloys, Phase diagrams, Thermodynamics

I. INTRODUCTION

Monte Carlo (MC) methods are the workhorse tool for studying the phase diagrams, thermodynamics and phase equilibria of a wide range of atomistic/lattice models of materials [1]. However, a longstanding limitation of standard MC methods, based on the Metropolis algorithm [2], is their restriction to obtaining results for only a single simulation temperature for a given simulation instance, and their inefficiency (particularly their slow rate of convergence) near discontinuous phase transitions. Near such discontinuous transitions, it is typically found that accurate characterisation of sharp features in state functions, such as peaks in specific heat, often requires a large number of independent simulations, each of which must utilise a substantial number of trial MC moves. To overcome this issue, several so-called ‘enhanced’ sampling techniques have been developed, including umbrella sampling, transition matrix MC, and Wang–Landau sampling [3–9].

The root cause of the inefficiency of conventional sampling techniques in some settings is the fact that, in complex materials, ergodicity can be severely limited. This is a particularly significant problem when large free energy

barriers separate regions of configuration space. Umbrella sampling (for example) addresses this by introducing a bias potential that facilitates transitions across these barriers, enabling more uniform sampling throughout the configuration space of the model. This method is part of a broader class of re-weighting techniques, in which the usual Boltzmann factor is replaced by a modified weight function, in order to remove the ‘real’ thermodynamics such that a more uniform sampling is obtained. From this (biased) more uniform sampling, the (unbiased) Boltzmann statistics are subsequently recovered during post-processing.

Building on this concept, the Wang–Landau (WL) [10–14] sampling algorithm can be used to obtain the density of states (DOS) of a given model in the canonical ensemble, which can then be used to compute physical properties such as specific heat and internal energy—both as functions of temperature—as a simple post-processing step. The WL algorithm samples configurations in inverse proportion to the DOS, an initial estimate of which is iteratively refined during the sampling process until a uniform sampling distribution in energy is obtained. In this way, from a single simulation, one obtains both a set of samples which span the entire energy domain of interest, and the DOS over that same energy domain. WL sampling has found utility across a range of research areas, including the study of phase transitions [15–17], polymer physics [18–20], solubility prediction [21, 22], magnetic materials [23–28], and surface adsorption (adhesion of atoms/ions/molecules to surfaces) [29–31]. Methods to update the statistical temperature of the system instead of the DOS within WL have also been proposed and used to develop an algorithm equivalent of WL for molecular dynamics known as statistical temperature molecular dynamics (STMD) [32].

* Corresponding Author

Hubert.Naguszewski@warwick.ac.uk

† christopher.woodgate@bristol.ac.uk

‡ D.Quigley@warwick.ac.uk

Various schemes have been proposed to parallelise the WL algorithm [33], and more broadly to parallelise other classes of multicanonical simulations [34, 35], with the aim of optimising utilisation of compute resource and minimising simulation runtime. However, there is no consensus on which scheme—or combination of schemes—is the most efficient and offers the most speedup per WL instance, where each instance corresponds to one CPU core in the context of the present work. (This is because evaluation of our model Hamiltonian is performed using a single CPU core.) A common feature among many proposed parallelisation strategies is the decomposition of the energy domain (the total range of energies accessible to the simulation system) into energy sub-domains (smaller windows of energy). The WL algorithm is then applied concurrently in each energy sub-domain, and subsequently these are combined to produce the DOS across the entire energy range of interest.

This paper aims to identify the most efficient parallel implementation of the WL algorithm, with a particular focus on iterative runtime load balancing. The context of our work is the need to perform a great many high throughput WL calculation with varying Hamiltonian parameters and material compositions. We are hence primarily concerned with maximising per-core parallel efficiency rather than reaching the highest possible speedup or core count for a single WL calculation. A key development of the present work is the demonstration that, despite the challenging nature of achieving perfectly optimal load balancing, it can be closely approached by dynamically adjusting the size of each energy sub-domain after each WL iteration, thereby leading to improved computational performance over long simulation runs. We also demonstrate how the performance of the algorithm can dramatically depend on several factors, including: the choice of the number of energy sub-domains and their size; the number of walkers within each energy sub-domain; and the size of the overlap between neighbouring energy sub-domains.

The remainder of this manuscript is structured as follows. First, in Sec. II A we give an overview of the WL sampling algorithm. Subsequently, in Sec. II B, we discuss the various means by which the algorithm may be parallelised. Following on from this, in Sec. II C we present the model system used for benchmarking—a high-entropy alloy—and how it was implemented, along with a brief reprise of metrics for assessing parallel efficiency with respect to number of sub-domains and walkers. Proceeding in Sec. III, we present results comparing the efficiency of parallelisation schemes in the context of a benchmark model system, both in isolation and in combination. Finally, in Sec. IV, we conclude with recommendations on which parallelisation scheme(s) to prioritise when implementing flat histogram MC methods. Throughout the paper we will note where we expect conclusions to be specific to the system and where they are more generally applicable.

II. METHODS

A. Wang–Landau (WL) Sampling

WL sampling [1, 10] is a flat energy histogram technique with broad applicability. For a given model system—with an associated Hamiltonian that defines the energy as a function of the model’s configuration—the algorithm determines the DOS, from which thermodynamic quantities can subsequently be calculated for any temperature. This is in contrast to the simpler Metropolis MC algorithm, in which a separate simulation is required at every discrete temperature of interest. In order to obtain the DOS, we must first define an accessible energy domain of interest. This is perhaps most easily found by performing a simulation using the Metropolis algorithm to determine the extrema of the energy at the highest and lowest temperatures of interest. Using the energies found as a guide, we can choose an energy domain which is uniformly divided into N bins of equal size, with each bin representing an energy macrostate containing multiple model system configurations. Given a model in which microstates are discrete—such as the lattice-based model in this study—the partition function Z is expressed as a sum over all microstates. This can be re-written as a sum over all the discrete energies E [1], *i.e.*

$$Z = \sum_i \exp(-E_i/k_B T) \equiv \sum_E g(E) \exp(-E/k_B T), \quad (1)$$

where E_i is the energy of the i^{th} microstate, k_B is the Boltzmann constant, T is the temperature, and $g(E)$ is the density of states corresponding to energy E . The expression on the right of Eq. 1 is then approximated and re-written as a sum over ‘binned’ energy macrostates, taking the form

$$Z \approx \sum_j g(E_j) \exp(-E_j/k_B T), \quad (2)$$

where E_j is the energy of the centre of the j^{th} energy bin, and $g(E_j)$ is the DOS corresponding to the j^{th} energy bin. (Note that the first summation appearing in Eq. 1 is the usual summation over microstates, and that the second summation of Eq. 1 is taken over *all* energies accessible to the system. This is in contrast to the approximate summation of Eq. 2, which is a sum over the newly-defined energy bins, taken to represent macrostates.)

In order to obtain the DOS, the WL algorithm performs a random walk in the configuration space of the model. Trial moves from an initial configuration, labelled n , to a proposed configuration, labelled m , are accepted with probability

$$P_{n \rightarrow m} = \begin{cases} \frac{g(E_n)}{g(E_m)}, & g(E_n) < g(E_m) \\ 1, & g(E_n) \geq g(E_m), \end{cases} \quad (3)$$

where E_n is the energy of the initial configuration and E_m is the energy of the proposed configuration. In the

context of the fixed-lattice high-entropy alloy simulations considered in this work, MC trial moves consist of two randomly selected atoms within the system swapping positions, accepted according to the probability in Eq. 3. This algorithm is not self-starting however, as the DOS is not known at the start of the simulation and an initial estimate for the DOS is required. This is usually taken as $g(E) \equiv 1$, and is iteratively refined throughout the simulation, with the uniformity of sampling being a measure of how accurately the DOS is currently estimated. Once the DOS is sufficiently accurate, a uniform distribution of samples is obtained. The estimate of the DOS is refined in iterations by modifying the DOS after each trial move according to

$$g(E_i) \rightarrow g(E_i)f_k, \quad (4)$$

with E_i being the energy of the resulting state, while f_k is known as the modification factor at WL iteration k , which in practice takes a value close to, but slightly greater than, unity. (Note that this means that there are no ‘wasted’ moves; even a rejected move results in the DOS being updated.) Due to the many orders of magnitude over which the DOS varies, it is often numerically convenient to work with $\log g(E)$ rather than $g(E)$ itself, incremented by $\log(f_k)$ at each trial move. To further address the issue of magnitude, the logarithm of the DOS is periodically shifted such that the minimum value is zero. At the start of the simulation, the iteration number is $k = 1$ and $\log(f_1)$ is set to some appropriately small value—we choose $\log(f_1) = 0.05$. A histogram of the visited energies, $H(E)$, is maintained, and the WL iteration number is incremented once the histogram reaches a desired level of ‘flatness’, F , defined as

$$F = \frac{\min(H(E))}{\frac{1}{N} \sum_i H(E_i)}, \quad (5)$$

where the target value of F is typically chosen to be close to 0.8 (80%), indicating that the simulation is now exploring all energy windows near-equally. If this flatness condition is met, the sampling is paused, and the modification factor f is reduced to be closer to unity for the next iteration, *e.g.* $f_{k+1} = \sqrt{f_k}$. The histogram, $H(E)$, tracking the visited energies is reset to be uniformly zero, and the sampling continues until f falls below a set tolerance $f = 1 + \varepsilon$ for some appropriately small value of ε , defining the degree of convergence of the DOS. The DOS is then taken to be that obtained after the final WL iteration. In this work, we take $\varepsilon = 2 \cdot 10^{-5}$ purely as a convenient level of convergence to use as a benchmark. In practice, ε could be taken as much smaller than this for generating accurate results, and convergence of any calculated output quantities for a given model system should always be thoroughly tested.

Once the DOS for the system is obtained, the energy probability distribution at a particular temperature can be calculated as a simple post-processing step,

$$P(E_i, T) = \frac{g(E_i) \exp(-E_i/k_B T)}{Z}, \quad (6)$$

where Z is calculated according to Eq. 2. From this distribution, various system properties can then be derived. An example of such a property is the specific heat, given by

$$C = \frac{\langle E^2 \rangle - \langle E \rangle^2}{k_B T^2}, \quad (7)$$

where $\langle E^2 \rangle$ and $\langle E \rangle$ are the average values of the square of the simulation energy and the simulation energy at a particular temperature, respectively. These averages can be calculated using the energy probability distribution of Eq. 6. In this manner, it can be seen that, once a sufficiently accurate estimate of $g(E)$ has been obtained, it is possible to run a simulation which samples all energies and temperatures of interest without further modification of the DOS ($f = 1$). In principle, the distribution for any observable sampled during this phase of the simulation (*e.g.*, a choice of order parameter describing a phase transition) can also be re-weighted to calculate its value at any temperature of interest.

Independently of any parallelisation strategy, there have been efforts to optimise both the algorithm’s ‘modification factor’ and ‘flatness criterion’ [36–39]. The WL algorithm was quickly improved by the introduction of energy domain decomposition into sub-domains [10]. This decomposition allowed for independent random walks to be performed either in serial or concurrently. Further optimisation can be achieved by dynamically adjusting the size of each energy sub-domain, although achieving maximally optimal load balancing is challenging due to the complex nature of the free energy landscape for a given system, several methods have been proposed for the dynamic adjustment of sub-domain size within a WL simulation [40–42]. These methods typically rely on restricting the energy domain over which a single WL random walk occurs. Once the flatness criterion has been met on a subset of the total energy domain, that portion of the energy domain is excluded from the current available energy for sampling during the iteration. With the advent of more accessible multi-process computing, these methods can be further expanded by combining non-uniform energy sub-domains with adaptive energy sub-domain sizing, instead of just modifying the total sampling energy domain for a single WL instance.

B. Parallelisation Strategies

In the context of parallel simulations, as mentioned in Refs. [12, 13], it is possible to subdivide the total simulation energy domain into smaller energy sub-domains, each sampled by an independent WL simulation, also referred to as a WL instance. Within each sub-domain, multiple independent WL simulations can operate concurrently, with each separate simulation being referred to as a ‘random walker’ or ‘walker’. The combination of multiple sub-domains with potentially multiple walkers

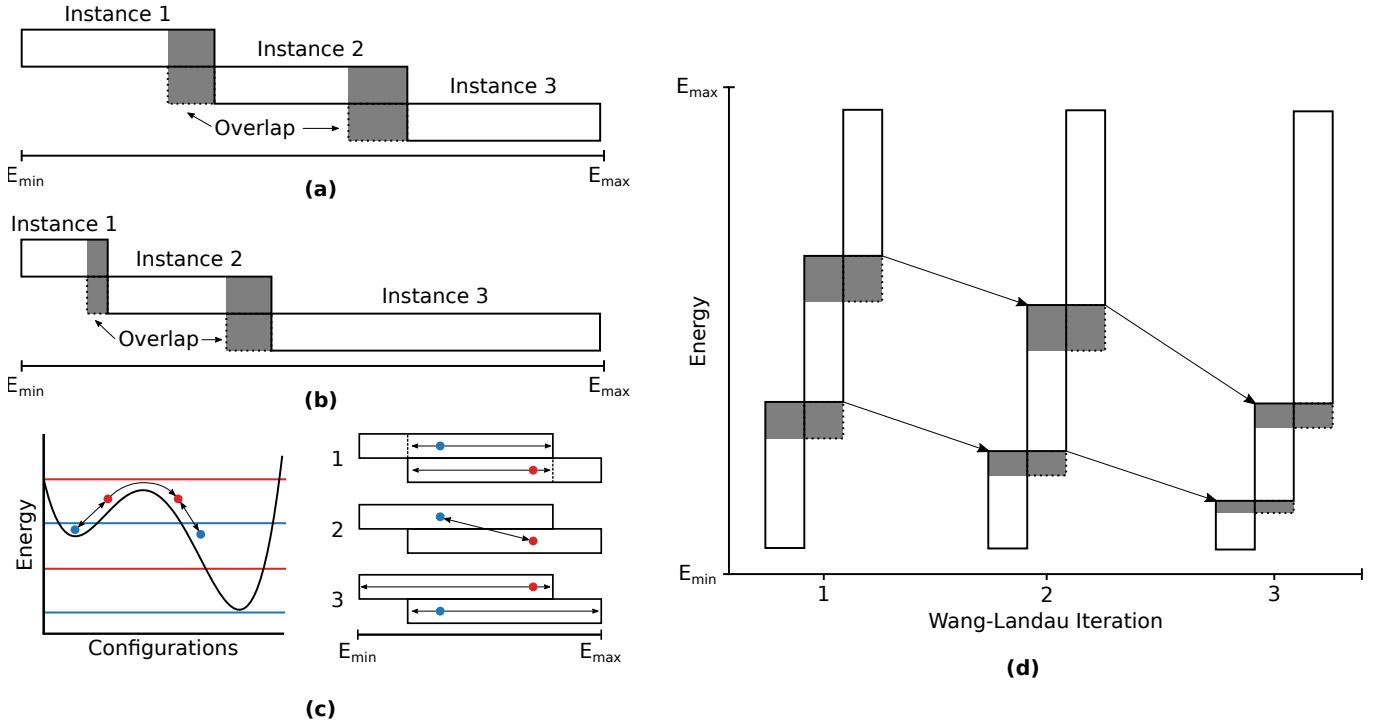


FIG. 1. Conceptual illustrations of the schemes discussed in this work for parallelising and/or accelerating parallel Wang-Landau sampling implementations. Panel (a) illustrates *uniform energy domain decomposition*, where the energy domain $[E_{\min}, E_{\max}]$ is evenly partitioned into sub-domains, with fixed percentage overlapping regions (shaded). Panel (b) illustrates *non-uniform energy domain decomposition*, where the energy domain $[E_{\min}, E_{\max}]$ is partitioned into non-uniform sub-domains, with fixed percentage overlapping regions (shaded). Panel (c) illustrates *replica exchange*, where independent walkers sampling within overlap regions can (occasionally) exchange configurations with neighbouring sub-domains (illustrated by red/blue particles), which allows for crossing configuration barriers. Finally, panel (d) represents *dynamic load balancing*, where energy sub-domains are adaptively adjusted after each Wang-Landau iteration based on the time taken to converge each sub-domain.

per sub-domain leads to many instances of the WL algorithm. The optimal distribution of a finite number of available instances (due to finite computational resource) to walkers and windows will be explored below. It is further possible to tune the sizes of the energy sub-domains by making them occupy unequal portions of the energy domain, which is particularly useful when there are significant differences between the total number of configurations available in different energy sub-domains. These parallelism approaches have been combined with replica-exchange MC (a method that facilitates configuration exchange between walkers in neighbouring energy sub-domains) [43–46] and studied previously [33].

Building on these ideas, all parallelisation schemes compared in this work make use of *energy domain decomposition* in some form. This involves subdividing the total energy domain considered in the simulation into smaller energy sub-domains, each of which is sampled by one or more independent WL instances, *i.e.* a random walker or set of walkers. In our implementation, each random walker constitutes an independent CPU process concurrently evolved throughout the simulation. A conceptual illustration of the two different energy domain decomposition schemes considered in this work is given

in panels (a) and (b) of Figure 1.

When the simulation begins, each energy sub-domain is assigned a walker (or set of walkers) and is initialised with a given physically relevant system configuration. (In the context of the alloys studied in this work, each walker is initialised with a random, lattice-based atomic configuration.) Frequently, these initial configurations have energies which lie outside the energy sub-domain assigned to the WL instance. Before sampling can begin, these walkers must be evolved so that they lie in the assigned energy sub-domain. In this work, this is achieved by running an MC simulation with an acceptance probability of

$$P_{n \rightarrow m} = \begin{cases} 1, & (E_m - E_T)^2 - (E_n - E_T)^2 \leq 0 \\ 0, & (E_m - E_T)^2 - (E_n - E_T)^2 > 0, \end{cases} \quad (8)$$

where E_T is the ‘central’ energy of the target sub-domain. In complex energy landscapes, it remains possible for a walker to become ‘trapped’ in a region of configuration space. To remedy this potential issue in the present study, if a walker is not within its specified energy sub-domain following a chosen number of trial MC moves, its atomic configuration is reinitialised to a random state and re-evolved towards its targeted energy using the acceptance probabilities defined in Eq. 8 once more. We

note that this approach functions in the case of the present work, but might not necessarily be broadly applicable. In other contexts with more complex energy landscapes it may be appropriate to apply a harmonic umbrella potential [47] to each walker to encourage it toward its assigned window during this initial “burn-in” phase of the simulation.

Each energy sub-domain overlaps with its adjacent energy sub-domains to facilitate combining the local DOS (the DOS associated with each energy sub-domain) into a global DOS across the whole domain. After each WL iteration has concluded, the global DOS is pieced together starting with the lowest energy sub-domain. The first and second energy sub-domains have their DOS combined by finding where the statistical temperature

$$\beta = \frac{d \log [g(E)]}{dE} \approx \frac{\log g(E + \Delta E) - \log g(E)}{\Delta E},$$

computed from each local DOS, best coincides within the overlap region. This is done by calculating β for each pair of neighbouring bins within the overlap region and then selecting the β that best coincides between the two neighbouring energy sub-domains as the joining point for the DOS. This process is repeated for each subsequent energy sub-domain until a pieced together DOS is obtained for the whole energy domain. If there are multiple walkers within an energy sub-domain, their individual DOS are averaged before the DOS are combined.

Due to the division of the energy domain into multiple sub-domains, and non-uniformity of simulation convergence times across these various sub-domains, additional modifications are made to determine when a WL iteration is finished. The modification factor remains constant across all sub-domains during an iteration and is only updated once the WL iteration ends. All walkers continue sampling with the current modification factor, even if they have satisfied the flatness criterion, until all walkers within the simulation also meet the flatness criterion. This avoids wasted CPU time by preventing WL instances from idling while ‘waiting’ for other walkers to converge.

1. Choice of Energy Domain Overlap Size

In the case of uniform energy domain decomposition, construction of energy sub-domains begins with uniformly dividing the energy domain into equal-sized sub-domains, after which the overlap is determined as a fixed percentage of the preceding sub-domain. This creates an overlap between an energy sub-domain and both of its neighbours.

The overlap calculation proceeds from lower to higher energy sub-domains. The first sub-domain is kept at its original size. Each subsequent sub-domain (sub-domain $i + 1$) is extended to include a fixed percentage of the updated size of the preceding sub-domain (sub-domain i), rounded up to the nearest integer number of energy bins.

In the present study, we always require a minimum overlap of 2 bins between neighbouring energy sub-domains to facilitate reliable construction of the global DOS.

2. Non-Uniform Energy Domain Decomposition

The time taken for each WL iteration is given by the time to converge of the slowest-converging sub-domain, since each walker must first converge their local (sub-domain) DOS histogram before the WL iteration can finish and create a combined global DOS. In general, the time taken to converge each sub-domain is non-uniform across the total energy domain, tending to take longer to converge around phase transitions or at low energy in the presence of degenerate microstates. In the former case, systems may exhibit entropic bottlenecks which are difficult to sample, and in the latter case the number of accessible configurations necessitates long sampling times. This non-uniform convergence time can be somewhat mitigated by altering the size of each WL energy sub-domain, however optimal load balancing is impossible *a priori*, as that requires the converged DOS states, which is the quantity being computed via simulation. By adjusting the individual size of each sub-domain, one can obtain an energy decomposition which is non-uniform in energy. This is done before the first iteration via a pre-sampling step, which evolves the WL algorithm until each energy bin across the entire energy domain has been visited by at least one walker per sub-domain a thousand times. Performing a pre-sampling step that is much longer than this is inefficient, since the simulation is progressing with sub-domain sizes which are non-optimal. This pre-sampling step is performed with uniformly sized sub-domains and lasts on the order of one WL iteration. The time taken per sub-domain is used to obtain the initial non-uniform energy domain decomposition. The details of the explicit means by which this initial non-uniform energy domain decomposition is obtained are discussed below, in Sec. II B 4. A conceptual illustration of a non-uniform energy domain decomposition scheme is given in Figure 1, panel (b).

3. Replica Exchange

Due to the often complex nature of the configuration space and associated energy landscape of the alloy being studied, replica exchange has been implemented within our test case to investigate its effectiveness in the context of the present work. Replica exchange allows for energy sub-domains to exchange walkers with neighbouring energy sub-domains provided that both configurations have energies which lie within the shared overlap region. This can, in principle, provide a way for walkers to reach potentially inaccessible regions within their assigned energy sub-domain by crossing energy barriers that require higher energies to access. This is anticipated to be par-

ticularly useful at low energies, where it could prevent walkers lying within a low-energy sub-domain from being trapped within a local minimum of the energy landscape. A conceptual illustration of replica exchange is given in Figure 1, panel (c). Within the model simulation, replica exchange swaps are proposed every 10 MC sweeps—where each MC sweep consists of a number of MC moves equal to the total number of atoms in the simulation cell—and the replica exchange swap is accepted based on acceptance probability of Eq. 3, *i.e.* the same acceptance probability as for any regular trial MC move.

4. Dynamic Load Balancing

As mentioned above, in Sec. IIB 2, perfectly optimal load balancing is anticipated to be impossible in general for arbitrary model systems. However, one can begin to approach optimal load balancing by iteratively adjusting the size of energy sub-domains after each complete WL iteration based on the time taken to converge the individual sub-domain visit histograms. A conceptual illustration of how such an iterative scheme might function is given in Figure 1, panel (d).

In this work, we propose a dynamic load balancing scheme which adjusts the sizing of energy sub-domains based on the run-time(s) of the walker(s) operating in that sub-domain at the end of the current WL iteration. This dynamic adjustment occurs after each WL iteration, and is carried out using the same algorithm as was proposed in the pre-sampling step for non-uniform energy domain decomposition, Sec. IIB 2. The number of energy bins, d_i , assumed to be the optimal number to be included in energy sub-domain, i , for a given iteration, k , is computed simply based on the relative MC steps taken to converge each sub-domain in that iteration according to

$$d_{i,k} = \frac{1/t_{i,k}}{\sum_i 1/t_{i,k}} N, \quad (9)$$

where i is the index of the energy sub-domain, $t_{i,k}$ is the number of MC steps taken per bin for energy sub-domain i to reach the flatness criterion in iteration k , and N is the total number of energy bins across the entire energy domain.

Our proposed extension of this scheme also updates the sub-domain size at each subsequent WL iteration according to

$$d_{i,k+1} = a_k \cdot d_{i,k-1} + b_k \cdot d_{i,k} \quad (10)$$

$$a_k + b_k = 1$$

where $d_{i,k}$ is the size of sub-domain i after iteration k as a fraction of the total number of bins, and a_k and b_k are scaling factors. To ensure the entire energy domain is covered, the $\sum_i d_{i,k+1} = N$. The purpose of a_k and b_k is to introduce memory (*i.e.* incremental changes to

previous sub-domain sizing) into the size of the energy sub-domains in order to reduce overcompensating after an outlier in convergence time. With each subsequent WL iteration k , b_k decreases such that the sub-domain adjustment decreases with each iteration. Explicitly, in this work, a_k and b_k were chosen according to

$$\begin{aligned} a_k &= 1 - b_k, \\ b_k &= \alpha \cdot \beta^k, \\ a_0 &= 1 - b_0, \\ b_0 &= \alpha \cdot \beta^0, \end{aligned} \quad (11)$$

where α controls the memory, and β controls the adjustment rate of sub-domain size. We find $\alpha = 0.2$ and $\beta = 0.9$ to be a reliable choice of parameters for use in the context of the present work. Of note, the initial update following the pre-sampling step is performed using a uniformly initialised d with $b = 1$. Additionally, in this work, to prevent zero-bin energy sub-domains, a minimum sub-domain size is imposed of 2% of the total number of bins across the entire energy domain.

After such a sub-domain adjustment, walkers may no longer be in their designated sub-domain and as such they are moved into their appropriate energy domain, as outlined at the start of this section, *i.e.* by performing (and restarting as necessary) an MC simulation with acceptance probability as defined in Eq. 8. (Note that accumulation of the histogram(s) and refinement of the DOS are paused during this adjustment process.) The sizes of energy overlap regions are also re-computed for the adjusted sub-domain sizing. We note here that a related approach has previously been explored, in which a single load balancing step was performed following the pre-sampling step [33], but where there was no further adjustment of energy sub-domain sizing following this initial load-balancing.

C. Model Implementation

As a test case for the parallelisation strategies studied in this work, we consider the application of WL sampling to the study of the phase behaviour of multicomponent, high-entropy alloys (HEAs). The application of WL to multicomponent alloys is an emerging area of research [48–50] and is therefore a relevant test case for this study.

HEAs are an intriguing class of disordered material, typically containing a mixture of four or more metallic elements combined in near-equal ratios, and are of interest for a range of next-generation technological applications as they have been shown to frequently possess superior physical properties when compared to traditional binary/ternary alloys. The large contribution made to the free energy of these materials by the configurational entropy is generally understood to stabilise disordered ‘solid solutions’ [51], in which there is a fixed, underlying crystal lattice, but where the constituent elements of the

material occupy these lattice sites in a random fashion. Several of such systems have been shown to exhibit both partial crystallographic ordering and/or phase decomposition with decreasing temperature. Computational modelling can shed insight into such diffusional solid-solid phase transformations. However, in the context of HEAs, computational simulation of phase equilibria faces several key challenges [52–55]. Primary among these is that the space of possible configurations (henceforth referred to as the ‘configuration space’) grows combinatorially both with the number of elements present in a considered composition, as well as with the number of atoms in a given simulation cell [56]. This means that a large number of configurations must be sampled for there to be confidence that results are well-converged and recovered quantities are representative of the relevant thermodynamic phases. WL sampling is hence extremely valuable in this context.

In this work, we consider the application of WL sampling to a fixed lattice model of the AlTiCrMo high-entropy alloy as a benchmark system, which is classified as a refractory high-entropy superalloy [57] and forms a (partially) chemically ordered structure with decreasing temperature [58, 59]. The parallelisation schemes considered in this study are implemented within the BRAWL package [60], which focuses specifically on fixed-lattice simulations of substitutional alloys. In particular, the code implements the Bragg–Williams model [61, 62], which describes the internal energy of a given alloy as a sum over atom-atom effective pair interactions, which are assumed to have finite range. Explicitly, this Hamiltonian takes the form

$$H(\{\xi_{i\alpha}\}) = \frac{1}{2} \sum_{i\alpha;j\alpha'} V_{i\alpha;j\alpha'} \xi_{i\alpha} \xi_{j\alpha'}, \quad (12)$$

where $V_{i\alpha;j\alpha'}$ denotes the effective pair interaction (EPI) between an atom of chemical species α on lattice site i and an atom of chemical species α' on lattice site j . $\{\xi_{i\alpha}\}$ indicates the site occupancies, where $\xi_{i\alpha} = 1$ if site i is occupied by an atom of chemical species α , and $\xi_{i\alpha} = 0$ otherwise. It is helpful to observe that this Hamiltonian can be thought of as a generalisation of the Lenz–Ising model used in elementary studies of magnetic phase transitions [63]. The EPIs of Eq. 12 are generally assumed to be homogeneous, isotropic, and of finite range, which simplifies evaluation of this expression. In this work, the EPIs used are those obtained in Ref. [59] for the AlTiCrMo HEA using the $S^{(2)}$ theory for multicomponent alloys [64–66].

D. Performance Metrics

In order to quantify the relative performance of the parallelisation schemes studied in this work, the following performance metrics are defined. First, we define the speedup, S , via

$$S(h, m) = \frac{N(1, m)}{N(h, m)}, \quad (13)$$

where $N(h, m)$ is the time taken to reach the flatness criterion by the slowest energy sub-domain, h is the total number of energy sub-domains, and m is the number of walkers per window.

We then define two metrics for measuring simulation efficiency. The first of these, ϕ_h , is used to quantify sub-domain efficiency and is defined as

$$\phi_h(h, m) = \frac{N(1, m)}{h \cdot N(h, m)}, \quad (14)$$

while the second, ϕ_m , is used to quantify walker efficiency and is defined as

$$\phi_m(h, m) = \frac{N(1, m)}{m \cdot N(h, m)}. \quad (15)$$

The sub-domain efficiency, ϕ_h , measures the mean speedup per sub-domain with a fixed number of walkers per domain, whereas the walker efficiency, ϕ_m , measures the mean speedup per walker at fixed number of sub-domains. In previous work [33], efficiencies of greater than 100% have been observed, and we expect the same to be observed here. Though initially counter-intuitive, efficiencies greater than 100% are possible in this context since not only does domain decomposition divide the work over concurrent CPU processes, it also has the potential to reduce the total number of energy evaluations required to converge the WL sampling simulation. This reduction in the work needed can be distinguished from traditional measures of efficiency in the context of CPU parallelism, since this reduction in work and hence speedup from sub-domain decomposition can be observed even on a single CPU sampling multiple sub-domains in order [41].

We can demonstrate approximately how the time taken for a sub-domain to converge is anticipated to scale with energy sub-domain size using a simple argument based on dimensional analysis. For a single walker, we model the energy sampled over time as a diffusive random walk in energy space. The time it takes for the probability distribution of finding the walker at energy E to evolve from an initial Dirac delta function to an eventual uniform distribution can be inferred from dimensional analysis as

$$t \propto D^\alpha \cdot L^\beta,$$

where t is the ‘time’ (*i.e.* number of MC trial moves) taken for a walker to diffuse across the sub-domain, D is the diffusivity and L is the size of the sub-domain. Using dimensional analysis one can obtain $\alpha = -1$ and $\beta = 2$ which leads to

$$t \propto \frac{L^2}{D}$$

from which we can see that $t \propto L^2$ for any fixed diffusivity, D . As such, the time taken for a walker to explore a sub-domain scales quadratically with the size of that sub-domain. In the case of h uniformly sized sub-domains the

Method Label		Parallelism Scheme(s)	
1	Dynamic Load Balancing	Non-Uniform Sub-domains	Replica Exchange
2	Dynamic Load Balancing	Non-Uniform Sub-domains	
3		Non-Uniform Sub-domains	Replica Exchange
4		Non-Uniform Sub-domains	
5		Uniform Sub-domains	Replica Exchange
6		Uniform Sub-domains	

TABLE I. Parallelism methods and associated labels used within the present study. All available combinations of the three separate parallelism methods are listed. Of note, it is not possible to have dynamic load balancing unless the energy sub-domains are non-uniform, as dynamic load balancing has non-uniform energy sub-domain sizing as a pre-requisite.

time taken, t_h , to converge is thus proportional to $(\frac{L}{h})^2$ per domain. Assuming that the same constant of proportionality holds for both the total energy case and the case of h uniform sub-domains, we have that $t_h/t = 1/h^2$, and it follows that

$$ht_h = \frac{t}{h}, \quad (16)$$

i.e. the total simulation time when the total energy domain is subdivided into h energy domains, ht_h , is reduced by a factor of h compared to the simulation time when running across the total sub-domain alone. Within this simple model, simulating each sub-domain on a separate CPU process can in principle result in a h^2 speedup and a sub-domain efficiency of $h \cdot 100\%$. However, this is a highly idealised model which does not account for overlaps between sub-domains, variation in diffusivity D across the energy domain and communication overheads. There is also an upper limit to h , beyond which there is an insufficient number of bins per sub-domain for the algorithm to function, motivating the use of multiple walkers. In principle however, the use of non-uniform sub-domain sizes should exactly compensate for the non-uniformity of D and hence it should be possible to approach the ideal speedup of h^2 provided the size of overlap regions, as well as practical details such as parallel communication overheads, are both minimised.

III. RESULTS AND DISCUSSION

In order to investigate the various outlined parallelisation schemes, the possible parallelisation schemes are grouped and labelled as ‘methods’, as shown in Table I. Once grouped, the effect of the schemes on speedup per WL instance, efficiency per sub-domain and efficiency per walker, was investigated over a range of h and m . The specific model system being simulated was equiatomic AlTiCrMo HEA, the phase behaviour of which has been investigated in an earlier work [59]. In this benchmarking study assessing the efficiency of various parallelisation schemes, we use a simulation supercell made up of $4 \times 4 \times 4$ bcc cubic unit cells for a total of 128 atoms. We find that

128 atoms is sufficient to represent the sampling challenge with tests at larger lattice sizes ($6 \times 6 \times 6$ bcc cubic unit cells, 432 atoms) confirming the trends illustrated here. Our WL sampling simulations used $N = 512$ energy bins over an energy domain of $E = -99$ meV/atom to $E = 0$ meV/atom, completing 12 full WL iterations starting with $\log f_1 = 0.05$, where each iteration used half of the logarithm of the modification factor of the previous iteration. The AlTiCrMo HEA model is well-suited as a benchmark system as it exhibits two phase transitions which occur at two separate temperatures [59], and hence convergence of the DOS is non-trivial. We investigated performance for a range of parameters within our parallelisation schemes with the number of walkers per sub-domain m varying from 1 to 6, the number of sub-domains h varying from 1 to 16 and the sub-domain overlap varying from 0% to 75% for both overlap schemes. It is important to note that we impose a requirement that a minimum of two bins must lie in the overlap region. As such, the case which we refer to as 0% overlap still has two bins in the overlap region. This minimum overlap does not change based on the sub-domain size.

Throughout, we plot results as a function of the total number of WL instances ($m \times h$) used in the calculation. We emphasise once again here that, in this work, evaluation of the model Hamiltonian (Eq. 12) is performed on a single CPU core, and hence the number of CPU cores strictly refers to the number of parallel WL simulation instances. If a more complex model Hamiltonian were to be studied, requiring multiple cores to be utilised for evaluation of the Hamiltonian, the total core count for a simulation would be $m \times h \times n_H$, where n_H was the number of cores used for evaluation of the model Hamiltonian. We therefore assert that our results will remain relevant for larger-scale simulations than those presented here.

Where error bars are presented on quantities, these are obtained by running five independent WL sampling simulations using different seeds on the pseudorandom number generator (PRNG) for the given choice of parallelisation scheme and associated settings. The data points represent the obtained average, while the error bars represent the standard deviation calculated across the five samples. We note that the default PRNG used

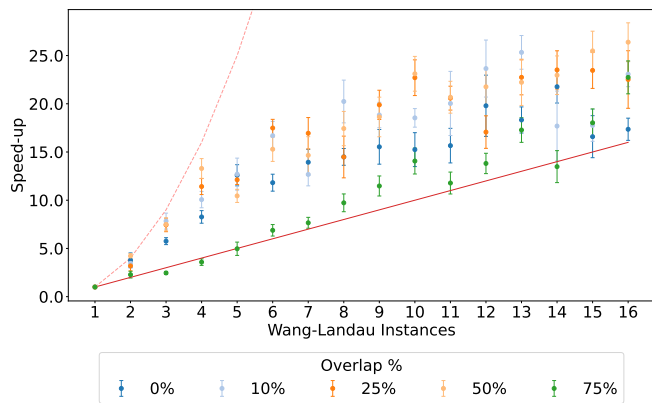


FIG. 2. Comparison of the speedup per WL instance for different choices of energy sub-domain overlap, taken relative to the case of a single walker across the entire energy domain. This plot is for simulations using Method 3 with a single walker per energy sub-domain. The dotted red line shows the square of the number of energy sub-domains, *i.e.*, the number of WL instances used squared (h^2), while the solid red line shows the number of energy sub-domains, *i.e.*, the number of WL instances (h). It can be seen that, for all choices of energy sub-domain overlap region size (excluding the 75% case), the speedup is significantly above 100% and that the choice of size of overlap region between energy sub-domains has noticeable impact on speedup.

in BRAWL [60] uses the Mersenne Twister algorithm [67], which has a long period and is well-suited to long sampling runs.

A. Effect of energy sub-domain overlap

We begin by considering the effect of energy sub-domain overlap. We choose to use Method 3 to illustrate this effect. We also choose to employ only a single walker per energy sub-domain, as we do not expect the number of walkers per sub-domain to change the impact of overlap region size on simulation speedup. Figure 2 compares the obtained simulation speedup as a function of number of WL instances employed for several choices of energy sub-domain overlap region size. It can be seen that the degree of overlap does not have a significant impact on the speedup achieved through WL sampling. Across all sub-domain counts, only in the 75% overlap case does the speedup begin to reduce significantly relative to the other choices of overlap region size.

Of the overlaps, the 25% to 50% overlap cases appears to be the ones with the most speedup, although not by a statistically significant amount. Such sub-domain overlap allows for better combining of the DOS, as well as a larger region for replica exchange, while still allowing walkers to focus more narrowly on their assigned sub-domain, resulting in more efficient sampling and higher speedup. The other cases introduce either too much or too little overlap, with too little overlap potentially harm-

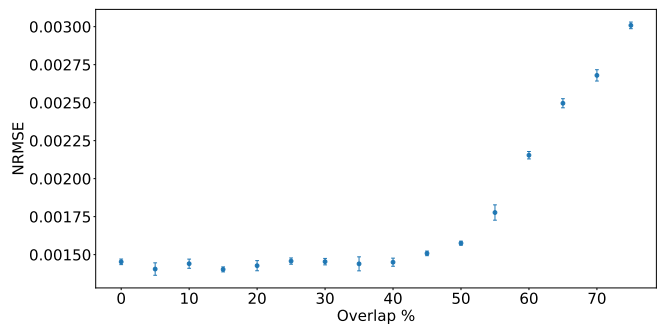


FIG. 3. Normalised root mean square error (RMSE) as a function of the percentage overlap. The RMSE was obtained by comparing 23 iteration sampling runs with 16 windows to a well-converged 36 iteration sampling run with 1 window. The error bars correspond to the standard deviation of time taken of 5 repeat WL simulations. For each number of overlap bins, 5 separate 23 iteration simulations were performed and averaged. These data show that there is no correlation between the size of the energy sub-domain overlap region and the accuracy of the obtained global DOS up to a 50% overlap. Beyond 50% overlap the averaging power of the sampling is reduced hence the increase in normalised RMSE though not to a statistically significant extent.

ing the DOS combining and too much overlap leading to a greater portion of the configuration space being sampled by multiple walkers, which reduces the benefits of parallelisation. For the remainder of this study a 25% overlap is used. This deliberate choice allows for evaluation of performance with an overlap width which remains optimal to within the statistical uncertainty of our data, but still provides a significant range of overlap to realise any benefit from including replica exchange.

It is important to note the super-linear scaling observed in some cases. This greater than 100% speedup per WL instance arises from the reduced number of total MC steps required when sampling due to the energy domain decomposition as explained in the previous section. This effect was discussed above, in Sec. IID, and is also examined further in Sec. IIID, below. Of further note is the speedup for the first 3 WL instances matches the theoretical limit of h^2 , derived in Sec. IID to within the uncertainty of our measurements.

To assess the effect of overlap on DOS convergence, in Figure 3 we show the normalised root mean square error (RMSE) between a 36 iteration (well-converged) WL simulation in one sub-domain and 23 iteration WL simulations of varying overlap size with 16 sub-domains. It can be seen that there is no correlation between DOS convergence accuracy and the percentage overlap in the energy sub-domain overlap region up to 50%. Beyond this point the averaging power of the overlap regions is decreased resulting in an increased normalised RMSE. Of note, however, is that this increase in normalised RMSE increases from 0.15% to 0.3% which is not a statistically significant increase. While remaining at or below a 50% overlap, that the primary benefit of the energy

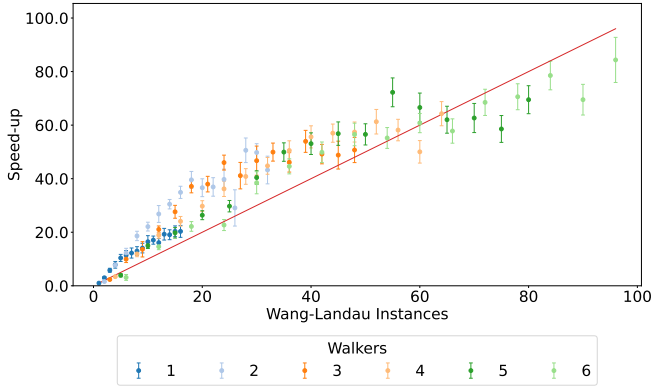


FIG. 4. Comparison of the speedup per WL instance for different numbers of walkers per energy sub-domain, taken relative to the case of a single walker across the entire energy domain. This plot is for Method 3 with a 25% energy sub-domain overlap. The error bars correspond to the standard deviation of time taken of 5 repeat WL simulations with different seeds on the PRNG. These data show that the optimal choices for efficiency are the 1 and 2 walker cases, with the 2 walker case marginally increasing the efficiency of each sub-domain. Additional walkers beyond this appear to confer no further benefit.

sub-domain overlap region(s) is to facilitate replica exchange and does not harm the sampling.

B. Influence of number of walkers used per energy sub-domain

Proceeding, we now consider the effect of including more than one walker per energy sub-domain. A natural choice for reducing calculation time in WL is to increase the number of walkers that sample the configuration space, as seen in Ref. [33] where 16 walkers are used per sub-domain. However, evidence of the resulting performance gains has not generally been reported, and we investigate this aspect explicitly here.

Figure 4 shows a comparison of the speedup per WL instance (taken relative to one walker per energy sub-domain) for different numbers of walkers per energy sub-domain for Method 3, *i.e.* using statically-sized non-uniform energy sub-domains. It can be seen that useful speedup can be obtained, which is across all walker cases. In all cases, the efficiency remains super-linear for the majority if not for all sub-domains, however going beyond around 60 instance the efficiency drops below 100% while also appearing to plateau in terms of speedup. The initial improvement from the walkers can be attributed to the Central Limit Theorem [68]. By increasing the number of walkers, the number of samples increases (*e.g.* 2 walkers give twice as many samples, *etc.*) which leads to a reduction in statistical error on the estimate of the DOS proportional to $1/\sqrt{n}$, where n is the number of samples. As the number of walkers increases this reduc-

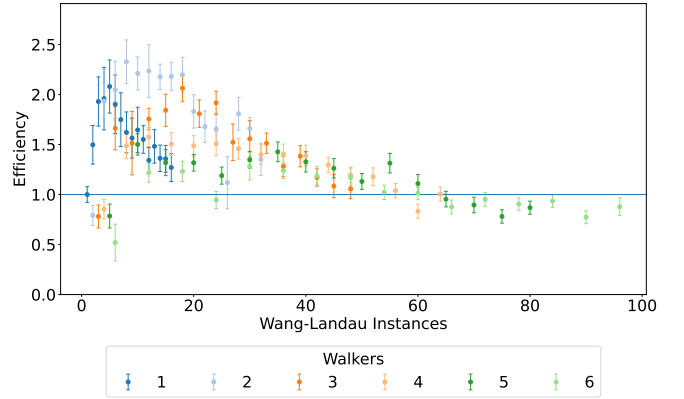


FIG. 5. Comparison of the walker efficiency per WL instance for different numbers of walkers per energy sub-domain, taken relative to the case of a single walker across the entire energy domain. This plot is for Method 3 with a 25% energy sub-domain overlap. The error bars correspond to the standard deviation of time taken of 5 repeat WL simulations with different seeds on the PRNG. The plot shows the efficiency of each walker introduced across 3 methods, with the 1 and 2 walker cases displaying the highest maximum efficiency per Wang-Landau instance.

tion in error goes as $1/\sqrt{2}$, $1/\sqrt{3}$, *etc.* which results in a quickly diminishing improvement in efficiency with respect to additional walkers. Our single sub-domain case with a varying quantity of independent walkers can be compared to the implementations found in Refs. [34, 35], however we only share statistics (*i.e.* the visit histogram and DOS) every 50 sweeps.

We then consider the efficiency of the walkers, as defined in Eq. 15 above. Figure 5 shows the efficiency of the instances seen in Figure 4. After the introduction of an additional walker, the maximum efficiency exceeds that of the single walker case, however going beyond two walkers per sub-domain, the efficiency begins to decrease substantially which would align with the previously outline Central Limit Theorem explanation. This increased efficiency from walkers stems partially from the minimum sub-domain size. Since the convergence time is limited by the slowest sub-domain, once the slowest sub-domain reaches the smallest allowed size, further addition of sub-domains will not result in increased efficiency. For the remainder of this study, all results presented are obtained using one walker per energy sub-domain.

C. Choice of method for construction of energy sub-domains

Next, we consider the importance of the use of energy sub-domains. Figure 6 shows the sub-domain efficiency relative to a single walker operating within the entire energy domain. Within this plot we can see the sub-domain efficiency for Methods 1, 3 and 5 (from left to right respectively) across a range of sub-domains. When

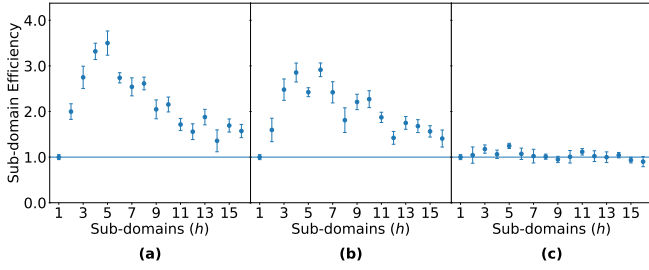


FIG. 6. Comparison of the energy sub-domain efficiency relative to the case of a single walker exploring the entire energy domain. Each method uses one walker per energy sub-domain and a 25% energy sub-domain overlap, and in all cases replica exchange was enabled. The error bars correspond to the standard deviation of time taken of 5 repeated WL simulations with different seeds on the PRNG. Panel (a) shows Method 1, *i.e.* dynamically load-balanced non-uniform sub-domains. Panel (b) shows Method 3, *i.e.* statically-sized non-uniform sub-domains. Panel (c) shows Method 5, *i.e.* uniformly-sized sub-domains. The plot shows the efficiency of each sub-domain introduced across 3 methods, with Method 1 showing the greatest and longest-lasting efficiency with its efficiency peak occurring at a smaller number of sub-domains. For the case of uniformly-sized energy sub-domains (Method 5), the sub-domain efficiency remains close to, or slightly below, unity regardless of the number of energy sub-domains used.

the three options for sizing of the energy sub-domains are considered, it can also be seen that the greatest increase in terms of performance is gained by implementing non-uniform energy domain decomposition. By further augmenting the sub-domains and introducing load balancing, a greater efficiency can be maintained at a higher h as compared to using non-uniform energy domain decomposition with static sub-domain sizes. The performance gain observed from energy domain decomposition can be attributed to a reduction in the time taken for any walker to diffuse across the sub-domain, which reduces wasted sampling time spent on easily converged bins. Further discussion of this aspect can be found in Sec. III D.

To demonstrate the efficacy of our proposed dynamic energy sub-domain sizing scheme, we plot the comparative sizes of the energy sub-domains for an illustrative simulation run in Figure 7, for the case where a total of six energy sub-domains were used. The first iteration here corresponds to the pre-sampling step with initial uniform domain decomposition. It can be seen that there is a difference in the size of the sub-domains at iteration 2 as compared to iteration 13. This demonstrates the difficulty of *a priori* load balancing in the pre-sampling step and the benefits conferred by iterative load balancing. Also of note is the fact that the relative sizes of domain settle over the course of many iterations, demonstrating the numerical stability of our proposed load-balancing scheme.

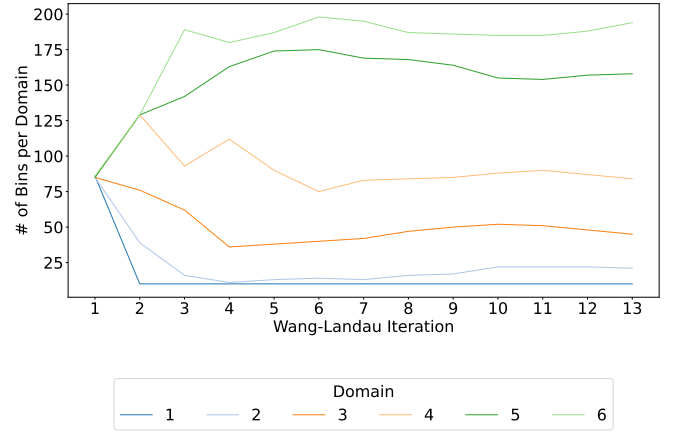


FIG. 7. A plot of the relative sizes of energy sub-domains for the proposed dynamic load-balancing scheme as a function of WL iteration number, to demonstrate the effect of load balancing on sub-domain size over the duration of a simulation. This plot is for Method 1 with one walker per energy sub-domain, a 25% energy sub-domain overlap, and a decomposition of the total energy domain into 6 sub-domains. The first iterations shows the initial uniform domain decomposition. It can be seen that the load balancing algorithm continues to make substantial adjustments to the relative sizes of energy sub-domains for the first few WL iterations, and that adjustment following an initial pre-sampling step is not enough to achieve good load balancing.

D. Origins of observed super-linear efficiency

Now that greater than 100% efficiency for energy sub-domain decomposition has been demonstrated (evidenced in Figures 2 and 4), we support our initial analysis given in Sec. II D and confirm that super-linear efficiency results in a reduction in the total number of MC steps needed (summed over all WL instances) to converge the DOS. This is broadly equivalent to the statement that division of the total energy domain into sub-domains—and restricting walkers to operate in these sub-domains—reduces the total number of MC steps required to adequately explore the configuration space for the construction of the DOS.

Figure 8 shows the relative total number of MC steps required to complete all WL sampling iterations as a function of WL instance count, *i.e.* number of energy sub-domains, as compared to the case of a single walker operating across the entire energy domain. It can be seen that the total number of MC steps required to converge the simulation substantially drops once energy sub-domains are introduced. For the case of uniform energy sub-domains (panel (c)), as the number of sub-domains increases the total number of MC steps (sum of all steps taken across all $m \cdot h$ walkers) increases. In our conceptual model of a diffusive random walk in energy, this lack of reduction in MC steps required is due to the non-uniformity of ‘diffusivity’, D , quantifying the speed of walker exploration across the energy domain. In turn,

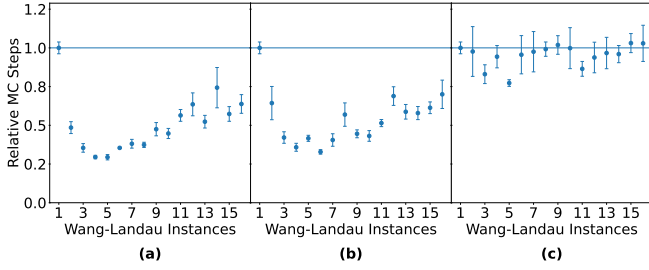


FIG. 8. MC steps taken to complete all WL sampling iterations relative to a single walker across the entire energy domain. Each method uses one walker and 25% overlap. Panel (a) shows results for Method 1, *i.e.* dynamically load-balanced non-uniform sub-domains. Panel (b) shows results for Method 3, *i.e.* statically-sized non-uniform sub-domains. Finally, panel (c) shows results for Method 5, *i.e.* uniformly-sized sub-domains. The plot shows that domain decomposition reduces the overall number of MC steps required to converge the WL sampling simulation.

this results in the WL convergence being dominated by the sub-domain which has the lowest D . However, for the two cases in which non-uniform energy sub-domains are employed (panels (a) and (b)), the total number of MC steps needed to fully converge the simulation decreases, which is particularly prominent for the first few additional WL instances, *i.e.* energy sub-domains. In this case, not only is the total work across the entire energy domain divided across WL instances, the total amount of work to be done is reduced (by around a factor of two for the case of two WL instances, *i.e.* energy sub-domains), which broadly agrees with the $\sim 4\times$ speedup seen in particular towards the left-hand data points of Figure 9. This analysis confirms our earlier assertion that the super-linear speedup, observed in both this study and in an earlier work, can be concretely understood in terms of a reduction in the total number of MC steps, *i.e.* the work, required to converge a given simulation.

E. Results summary

To summarise the key findings of our study, in Figure 9 we compare the overall simulation speedup (the most relevant metric for parallel performance of an algorithm for applications) across all six of the studied methods, *i.e.* parallelisation schemes. Here, all data points presented are for simulations run using a 25% energy sub-domain overlap and one walker per energy sub-domain.

It can be seen that all of the proposed methods achieve an appreciable speedup, with even the worst-performing methods (5 and 6) achieving near linear speedup out to 16 WL instances, *i.e.* energy sub-domains. However, the methods which stand out are 1, 2, 3, and 4, *i.e.* those using non-uniform energy domain decomposition. This result reinforces the idea that the greatest gain in performance is obtained via non-uniform domain

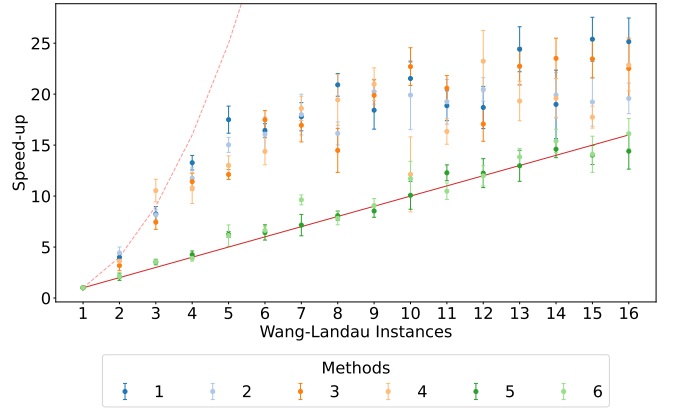


FIG. 9. Comparison of the speedup per WL instance, taken relative to the case of a single walker across the entire energy domain. The dotted line shows sub-domains squared (h^2). Each method has one walker and 25% overlap. The plot shows the greatest increase in performance can be obtained through non-uniform domain decomposition with load balancing further increasing performance.

decomposition. Methods 1 and 2, which make use of the proposed dynamic energy sub-domain sizing scheme, offer modest improvements over the case of statically-sized non-uniform energy sub-domains. The increase in performance between the load-balanced (Methods 1, 2) and not load balanced (Methods 3, 4) allows for the simulation to retain its sub-domain efficiency for a greater number of sub-domains giving access to greater speedups. Of note, replica exchange does not appear to significantly impact performance either positively or negatively.

IV. SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS

In summary, several parallelisation strategies for the WL sampling algorithm have been presented, implemented, and their performance assessed. The schemes studied, both in isolation and in combination, included energy domain decomposition, replica exchange, and a dynamic load-balancing scheme proposed in this work. From the performance evaluation within this study, several important insights emerge for the effective implementation of parallel WL sampling, which should—in principle—be extensible to other flat-histogram methods such as transition matrix MC [9].

The most significant factor in determining parallel performance is the choice of energy domain decomposition scheme, with use of non-uniform energy sub-domains consistently providing the largest gains in both speedup and overall simulation efficiency. Dynamic load balancing offers a modest further enhancement in performance by allowing the energy domain decomposition scheme to adjust itself ‘on the fly’ and adapt to the sampling demands of complex energy landscapes, with energy sub-

domain boundaries shifting between each WL iteration. Careful attention to the number of walkers allocated to each energy sub-domain also proves critical. While employing multiple walkers may seem like a natural choice when seeking to parallelise the WL sampling algorithm, our results indicate that this produces rapidly diminishing returns as a function of number of walkers, and potentially a reduction in efficiency. In this study, we found that the use of one or two walkers per sub-domain is generally sufficient to realise any performance improvements available through the use of multiple walkers per energy sub-domain. Also of note is that replica exchange does not noticeably influence efficiency in this test case, but nor does it compromise performance, which suggests that it may be included where needed to improve sampling without penalty. This is likely due to the non-physical MC moves (long-range atomic swaps) in our test system which result in the ergodicity not being limited by the need to cross energy barriers. Finally, the construction of overlap regions between sub-domains is shown to have a negligible effect on performance unless it is in excess of a 75% overlap, underscoring that large overlap is necessary only for replica exchange and not for the reliable combination of local DOS histograms to form the global DOS.

A. Recommendations

On the basis of these findings, we recommend that parallel implementations of WL sampling prioritise using non-uniform energy domain decomposition as a primary route to improved simulation efficiency. Subsequently, one should implement multiple walkers per sub-domain for further efficiency and performance. Once those two features have been implemented, dynamic load balancing should subsequently be incorporated wherever feasible for modest improvement of per-instance efficiency. If reduced runtime is more important than efficiency, and if adding further sub-domains hurts performance, further increasing the number of walkers can be employed as a complementary tool, though they will decrease computational efficiency. Finally, overlap regions of a modest size should be included primarily to facilitate replica exchange, as they have little to no impact on the accuracy of the reconstructed total DOS, though excessively-sized overlap regions should be avoided due to the associated drop in parallel performance.

CREDIT AUTHORSHIP CONTRIBUTION STATEMENT

Hubert J. Naguszewski Conceptualization (equal), Data curation, Investigation, Methodology (lead), Software (lead), Validation, Visualization, Writing - original draft (lead), Writing - review and editing (equal). **Christopher D. Woodgate** Methodology (supporting), Software (supporting), Supervision (supporting), Writing - original draft (supporting), Writing - review and editing (equal). **David Quigley** Conceptualization (equal), Funding acquisition, Project administration, Resources, Supervision (lead), Writing - review and editing (equal).

DECLARATION OF COMPETING INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

DATA AVAILABILITY

The data supporting the findings of this study can be made available upon request. The BRAWL package, which was used to obtain all results presented in this work is fully open-source and available at <https://github.com/ChrisWoodgate/BraWL>.

ACKNOWLEDGMENTS

H.J.N. acknowledges a funded studentship within the UK Engineering and Physical Sciences Research Council (EPSRC) Centre for Doctoral Training in Modelling of Heterogeneous Systems, Grant EP/S022848/1. C.D.W. acknowledges support from an EPSRC Doctoral Prize Fellowship at the University of Bristol, Grant EP/W524414/1. Calculations were performed using the Sulis Tier 2 HPC platform hosted by the Scientific Computing Research Technology Platform (SCRTP) at the University of Warwick. Sulis is funded by EPSRC Grant EP/T022108/1 and the HPC Midlands+ consortium. Additional computing facilities were provided by the Scientific Computing Research Technology Platform at the University of Warwick, and by the Advanced Computing Research Centre at the University of Bristol.

[1] D. P. Landau and K. Binder, *A Guide to Monte Carlo Simulations in Statistical Physics*, 5th ed. (Cambridge University Press, Cambridge, UK, 2021).

[2] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, *The Journal of Chemical Physics* **21**, 1087 (1953).

- [3] D. Frenkel and B. Smit, *Understanding molecular simulation: from algorithms to applications*, 2nd ed., Computational science series No. 1 (Academic Press, 2002).
- [4] C. H. Bennett, *Journal of Computational Physics* **22**, 245 (1976).
- [5] G. M. Torrie and J. P. Valleau, *Journal of Computational Physics* **23**, 187 (1977).
- [6] B. A. Berg and T. Neuhaus, *Physics Letters B* **267**, 249 (1991).
- [7] B. A. Berg and T. Neuhaus, *Physical Review Letters* **68**, 9 (1992).
- [8] J. Lee, *Physical Review Letters* **71**, 211 (1993).
- [9] J.-S. Wang, *Computer Physics Communications* **121-122**, 22 (1999), proceedings of the Europhysics Conference on Computational Physics CCP 1998.
- [10] F. Wang and D. P. Landau, *Physical Review Letters* **86**, 2050 (2001).
- [11] F. Wang and D. P. Landau, *Physical Review E* **64**, 056101 (2001).
- [12] D. P. Landau, S.-H. Tsai, and M. Exler, *American Journal of Physics* **72**, 1294 (2004).
- [13] D. P. Landau and F. Wang, *Computer Physics Communications Proceedings of the Europhysics Conference on Computational Physics Computational Modeling and Simulation of Complex Systems*, **147**, 674 (2002).
- [14] G. Brown, A. Rusanu, M. Daene, D. M. Nicholson, M. Eisenbach, and J. Fidler, *Journal of Applied Physics* **109**, 07E161 (2011).
- [15] N. Volkov, A. Lyubartsev, and L. Bergström, *Nanoscale* **4**, 4765 (2012).
- [16] J. H. Lee, S.-Y. Kim, and J. M. Kim, *Physical Review B* **109**, 064422 (2024).
- [17] K. Sato, S. Takizawa, and T. Mohri, *Journal of the Physical Society of Japan* **79**, 084602 (2010).
- [18] Z. Wang and X. He, *The Journal of Chemical Physics* **135**, 094902 (2011).
- [19] D. Antypov and J. A. Elliott, *The Journal of Chemical Physics* **129**, 174901 (2008).
- [20] D. F. Parsons and D. R. M. Williams, *The Journal of Chemical Physics* **124**, 221103 (2006).
- [21] S. Boothroyd, A. Kerridge, A. Broo, D. Buttar, and J. Anwar, *Physical Chemistry Chemical Physics* **20**, 20981 (2018).
- [22] S. Boothroyd and J. Anwar, *The Journal of Chemical Physics* **151**, 184113 (2019).
- [23] S. Bin-Omran, *Materials Research Bulletin* **95**, 334 (2017).
- [24] T.-L. H. Nguyen and V. Thanh Ngo, *Advances in Natural Sciences: Nanoscience and Nanotechnology* **8**, 015013 (2017).
- [25] S. Miyashita, M. Nishino, Y. Toga, T. Hinokihara, I. E. Uysal, T. Miyake, H. Akai, S. Hirose, and A. Sakuma, *Science and Technology of Advanced Materials* **22**, 658 (2021).
- [26] S. Bin-Omran, I. A. Kornev, and L. Bellaiche, *Physical Review B* **93**, 014104 (2016).
- [27] V. T. Ngo and H. T. Diep, *Physical Review E* **78**, 031119 (2008).
- [28] N. A. Bogoslovskiy, P. V. Petrov, and N. S. Averkiev, *Physical Review B* **109**, 024436 (2024).
- [29] M. P. Allen and A. D. Swetnam, *Physics Procedia Proceedings of the 25th Workshop on Computer Simulation Studies in Condensed Matter Physics*, **34**, 6 (2012).
- [30] C. Lazo and F. J. Keil, *Physical Review B* **79**, 245418 (2009).
- [31] A. D. Swetnam and M. P. Allen, *Physical Chemistry Chemical Physics* **11**, 2046 (2009).
- [32] J. Kim, J. E. Straub, and T. Keyes, *Physical Review Letters* **97**, 050601 (2006).
- [33] T. Vogel, Y. W. Li, T. Wüst, and D. P. Landau, *Physical Review Letters* **110**, 210603 (2013).
- [34] J. Zierenberg, M. Marenz, and W. Janke, *Computer Physics Communications* **184**, 1155 (2013).
- [35] J. Gross, J. Zierenberg, M. Weigel, and W. Janke, *Computer Physics Communications* **224**, 387 (2018).
- [36] C. Zhou and R. N. Bhatt, *Physical Review E* **72**, 025701 (2005).
- [37] R. E. Belardinelli and V. D. Pereyra, *Physical Review E* **75**, 046701 (2007).
- [38] C. Zhou, T. C. Schulthess, S. Torbrügge, and D. P. Landau, *Physical Review Letters* **96**, 120201 (2006).
- [39] A. D. Swetnam and M. P. Allen, *Journal of Computational Chemistry* **32**, 816 (2011).
- [40] A. G. Cunha-Netto and R. Dickman, *Computer Physics Communications* **182**, 719 (2011).
- [41] A. G. Cunha-Netto, A. A. Caparica, S.-H. Tsai, R. Dickman, and D. P. Landau, *Physical Review E* **78**, 055701 (2008).
- [42] A. G. Cunha-Netto, R. Dickman, and A. A. Caparica, *Computer Physics Communications Special issue based on the Conference on Computational Physics 2008*, **180**, 583 (2009).
- [43] C. J. Geyer, *Markov Chain Monte Carlo Maximum Likelihood*, Tech. Rep. Technical Report No. 568 (University of Minnesota, School of Statistics, 1991).
- [44] K. Hukushima and K. Nemoto, *Journal of the Physical Society of Japan* **65**, 1604 (1996).
- [45] A. Valentim, J. C. S. Rocha, S.-H. Tsai, Y. W. Li, M. Eisenbach, C. E. Fiore, and D. P. Landau, *Journal of Physics: Conference Series* **640**, 012006 (2015).
- [46] Y. Zhao, S. W. Cheung, Y. W. Li, and M. Eisenbach, *Physics Procedia Proceedings of the 27th Workshop on Computer Simulation Studies in Condensed Matter Physics (CSP2014)*, **57**, 43 (2014).
- [47] J. Kästner, *Wiley Interdisciplinary Reviews: Computational Molecular Science* **1**, 932 (2011).
- [48] Z. Pei, R. Li, M. C. Gao, and G. M. Stocks, *npj Computational Materials* **6**, 122 (2020).
- [49] K. Takeuchi, R. Tanaka, and K. Yuge, *Physical Review B* **96**, 144202 (2017).
- [50] Y. Nanba and M. Koyama, *Journal of Computer Chemistry, Japan-International Edition* **9**, 2022 (2023).
- [51] J.-W. Yeh, S.-K. Chen, S.-J. Lin, J.-Y. Gan, T.-S. Chin, T.-T. Shun, C.-H. Tsau, and S.-Y. Chang, *Advanced Engineering Materials* **6**, 299 (2004).
- [52] M. Widom, *Journal of Materials Research* **33**, 2881 (2018).
- [53] M. Eisenbach, Z. Pei, and X. Liu, *Journal of Physics: Condensed Matter* **31**, 273002 (2019).
- [54] A. Ferrari, B. Dutta, K. Gubaev, Y. Ikeda, P. Srinivasan, B. Grabowski, and F. Körmann, *Journal of Applied Physics* **128**, 150901 (2020).
- [55] A. Ferrari, F. Körmann, M. Asta, and J. Neugebauer, *Nature Computational Science* **3**, 221 (2023).
- [56] Y.-W. Zhang, V. Sorkin, Z. H. Aitken, A. Politano, J. Behler, A. P. Thompson, T. W. Ko, S. P. Ong, O. Chalykh, D. Korogod, E. Podryabinkin, A. Shapeev, J. Li,

- Y. Mishin, Z. Pei, X. Liu, J. Kim, Y. Park, S. Hwang, S. Han, K. Sheriff, Y. Cao, and R. Freitas, *Modelling and Simulation in Materials Science and Engineering* **33**, 023301 (2025).
- [57] D. B. Miracle, M.-H. Tsai, O. N. Senkov, V. Soni, and R. Banerjee, *Scripta Materialia* **187**, 445 (2020).
- [58] H. Chen, A. Kauffmann, S. Seils, T. Boll, C. Lieb-scher, I. Harding, K. Kumar, D. Szabó, S. Schlabach, S. Kauffmann-Weiss, F. Müller, B. Gorr, H.-J. Christ, and M. Heilmaier, *Acta Materialia* **176**, 123 (2019).
- [59] C. D. Woodgate, H. J. Naguszewski, D. Redka, J. Minár, D. Quigley, and J. B. Staunton, *Journal of Physics: Materials* **8**, 045002 (2025).
- [60] H. J. Naguszewski, L. B. Pártay, D. Quigley, and C. D. Woodgate, arXiv:2505.05393 .
- [61] W. L. Bragg and E. J. Williams, *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character* **145**, 699 (1934).
- [62] W. L. Bragg and E. J. Williams, *Proceedings of the Royal Society of London. Series A - Mathematical and Physical Sciences* **151**, 540 (1935).
- [63] S. G. Brush, *Reviews of Modern Physics* **39**, 883 (1967).
- [64] S. N. Khan, J. B. Staunton, and G. M. Stocks, *Physical Review B* **93**, 054206 (2016).
- [65] C. D. Woodgate and J. B. Staunton, *Physical Review B* **105**, 115124 (2022).
- [66] C. D. Woodgate, *Modelling Atomic Arrangements in Multicomponent Alloys: A Perturbative, First-Principles-Based Approach*, Springer Series in Materials Science, Vol. 346 (Springer Nature Switzerland, Cham, 2024).
- [67] M. Matsumoto and T. Nishimura, *ACM Transactions on Modeling and Computer Simulation* **8**, 3 (1998).
- [68] W. Feller, *An Introduction to Probability Theory and Its Applications, Vol. 1* (John Wiley & Sons, 1968).