AIP | Journal of Applied Physics

Ensemble averaging vs. time averaging in molecular dynamics simulations of thermal conductivity

Kiarash Gordiz, David J. Singh, and Asegun Henry

Citation: Journal of Applied Physics **117**, 045104 (2015); doi: 10.1063/1.4906957
View online: http://dx.doi.org/10.1063/1.4906957
View Table of Contents: http://scitation.aip.org/content/aip/journal/jap/117/4?ver=pdfcov
Published by the AIP Publishing

JOURNAL OF APPLIED PHYSICS **117**, 045104 (2015)

Ensemble averaging vs. time averaging in molecular dynamics simulations of thermal conductivity

Kiarash Gordiz,[1,a)] David J. Singh,[2] and Asegun Henry[1,3]
[1]*George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332, USA*
[2]*Oak Ridge National Laboratory, Materials Science and Technology Division, Oak Ridge, Tennessee 37831, USA*
[3]*School of Materials Science and Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332, USA*

(Received 16 December 2014; accepted 17 January 2015; published online 29 January 2015)

In this report, we compare time averaging and ensemble averaging as two different methods for phase space sampling in molecular dynamics (MD) calculations of thermal conductivity. For the comparison, we calculate thermal conductivities of solid argon and silicon structures, using equilibrium MD. We introduce two different schemes for the ensemble averaging approach and show that both can reduce the total simulation time as compared to time averaging. It is also found that velocity rescaling is an efficient mechanism for phase space exploration. Although our methodology is tested using classical MD, the approaches used for generating independent trajectories may find their greatest utility in computationally expensive simulations such as first principles MD. For such simulations, where each time step is costly, time averaging can require long simulation times because each time step must be evaluated sequentially and therefore phase space averaging is achieved through sequential operations. On the other hand, with ensemble averaging, phase space sampling can be achieved through parallel operations, since each trajectory is independent. For this reason, particularly when using massively parallel architectures, ensemble averaging can result in much shorter simulation times (~100–200X), but exhibits similar overall computational effort. © *2015 AIP Publishing LLC*. [http://dx.doi.org/10.1063/1.4906957]

I. INTRODUCTION

The objective in many atomistic simulations is to calculate structural, transport, kinetic, or thermodynamic properties from the microscopic dynamics of atomic/molecular motions and interactions. Collecting this information during the atomistic simulation is referred to as phase space sampling.[1,2] Phase space sampling can be done by two methods: time averaging and ensemble averaging. Time averaging is the sequential probing of phase space in which one trajectory (i.e., the evolution of a single microstate with time under Newtonian law) moves through phase space until a sufficient volume has been probed and a representative average has been obtained. In contrast, ensemble averaging probes the phase space with independent trajectories. Utilizing parallel processing,[3] these independent trajectories can be simulated concurrently, which can greatly reduce the time required for the user to collect the phase space data and retrieve the results. The results of statistical mechanics derivations are most often written in terms of ensemble averaging. However, if a dynamical system is ergodic the ensemble average becomes equal to the time average.[4] Calculation of properties that depend explicitly on the system dynamics, such as transport coefficients from linear response theory (e.g., mass diffusivity and thermal conductivity) requires one to track a trajectory in time to sample events that take a finite amount of time to occur (i.e., diffusion hops, phonon scatterings, etc.). Consequently, for such properties there is a minimum simulation time needed to gather data about the particular phenomena of interest. Monte Carlo (MC) methods, on the other hand, utilize ensemble averaging in its purest form by consecutively examining distinct individual snapshots of the system. As a result, MC is not applicable to the calculation of such dynamical properties, and time averaging is not interchangeable with ensemble averaging in such cases.

In this study, we examine and compare time averaging and ensemble averaging for molecular dynamics (MD) simulations to quantify their respective benefits in the calculation of thermal conductivity. We delineate the respective advantages of each approach by using two different metrics for evaluation: (1) elapsed time (ET), which is the time experienced by the user (i.e., in hours) before the results are obtained, and (2) the computational cost (CC), or the total computational effort expended (i.e., in processor*hours) to obtain the results. To generalize the results in the ensuing analysis such that the computational times are not specific to the hardware used, we use the total number of MD time steps to represent the ET. The actual ET for a user depends on the architecture, efficiency of the MD code, and various other factors. In addition, to enable a straightforward comparison independent of the specific hardware or code used, we use the product of the number of atoms being simulated multiplied by the number of MD time steps evaluated to quantify the CC. By presenting our data in this way, our conclusions remain independent of the specific hardware or software configuration.

[a)]Author to whom correspondence should be addressed. Electronic mail: kgordiz3@gatech.edu

Despite the wide application of ensemble averaging to the study of the structural properties of bio-systems (e.g., protein-folding[5,6] and biopolymers[7]), to the best of our knowledge, time averaging has been the most prevalent approach in the context of classical MD simulations, especially thermal conductivity calculations.[8–18] In time averaging, typically the property of interest is calculated from sufficiently long simulation times until convergence is achieved, which is when the result no longer changes significantly with increased simulation time (phase space sampling).[1] Very often, a combination of time averaging supplemented by several trajectories (on the order of 10) is used to achieve convergence. In some cases, when convergence is difficult, additional trajectories can be used to improve averaging. For instance, Chalopin *et al.* used forty trajectories[19] to smooth the oscillations present in the MD simulations of thermal conductance. However, to the best of our knowledge, we are unaware of a case where ensemble averaging has been used solely for the purpose of reducing the ET for the user. The rarity in ensemble averaging utilization may be partially due to the increased complexity on the part of the user. For time averaging, collected data are usually tabulated as a single contiguous file for each independent trajectory and independent trajectories are often kept in separate directory structures. For small numbers of independent trajectories, aggregating the data is straightforward and easy to manage by manual commands. For large numbers of trajectories, however, this can be more cumbersome, as some independent trajectories may fail to finish the time averaging (i.e., the simulation time exceeds the requested wall-time in a high-performance computing cluster, or an unexpected memory error), which complicates the aggregation in the end. Moreover, using time averaging in classical MD simulations, results are usually obtained in a reasonable time frame, which does not create a need for other approaches. As a result, time averaging has prevailed as the method of choice because it can be more straightforward to implement and it yields the same final result for systems that behave ergodically.

Reducing the ET for the user is particularly important for simulations such as first-principles MD (FPMD).[20–23] However, it is important to note that similar to classical MD, FPMD simulations do not reproduce quantum statistics, which is particularly important at low temperatures.[24] Nonetheless, the processor time for each time step in FPMD simulations is several orders of magnitude larger than that of classical MD based on empirical potentials,[25] and as a result time averaging can become rate limiting. If the system is ergodic, to gather the same amount of phase space data, rather than a single trajectory, we can launch multiple trajectories in different positions in phase space. If we have sufficient computing resources, we can assume that simulations for these multiple trajectories can be executed concurrently using parallel processing architectures. Thus, the same amount of information can be gained by shorter simulation times devoted to each individual trajectory, resulting in a significant reduction in the ET to retrieve the results. Likewise, for FPMD simulations, replacing the sequential calculation of time steps needed for time averaging with the concurrent calculations of multiple trajectories in parallel can drastically reduce the ET experienced by the user. Although the ET is reduced, it is not clear *a priori* if ensemble averaging will require more, less, or similar CC. To examine and compare the CC and the efficacy of time averaging and ensemble averaging methods in phase space sampling, here, we employ equilibrium MD simulations to calculate the thermal conductivity of solid argon and silicon. To make our conclusions independent of the materials chosen, we use two different materials described by two different inter-atomic potentials. Although our examples are based on classical MD simulations, the presented ideas and methods can be applied and will likely offer greater utility for more expensive simulation methods such as FPMD.

It is interesting to note that recent advances in the implementation of graphics processing units (GPUs) in the calculation of thermal conductivity using MD and the Greek-Kubo formalism[11,26] have shown speed up factors between 10 and 11.[11] However, in these reports, the main approach is still based on time averaging. These implementations incorporate the parallel structure inside GPUs to reduce the time needed to complete each time step for one trajectory as compared to central processing units (CPUs). We believe that having access to a cluster with a large number of GPU nodes could decrease the ET even more, but the increased speed would likely apply equally to time and ensemble averaging.

In Secs. II–V, the distance between trajectories is evaluated by two independence parameters introduced in Sec. III. Sections IV and V show the results and discussion for utilization of the techniques presented, and we show example of calculations for the thermal conductivities of solid argon and silicon structures, respectively. Finally, concluding remarks are presented in Sec. VI.

II. TRAJECTORY GENERATION SCHEMES

One of the barriers to the implementation of ensemble averaging is the generation of independent trajectories, to probe the relevant portions of phase space. The correct choice of trajectories can minimize the total number of trajectories needed for convergence. To conceptually understand this process, one could imagine generating a set of trajectories that are localized in a very small region of the phase space. These localized trajectories cannot probe the phase space efficiently, so more trajectories are required to achieve convergence because each trajectory adds little information to the statistical average. In contrast, choosing a group of trajectories equally spaced over the relevant phase space volume being probed can minimize the number of trajectories needed to reach convergence, as each trajectory would add the maximum information to the statistical average. From this perspective, the generation of trajectories can be critical. In trajectory generation, the goal is to produce trajectories that are equidistant on the relevant hyper surface in phase space and equilibrate them with the minimum CC. To avoid highly improbable atomic configurations, the process of equilibration is indeed important as data gathered in this regime can significantly impact the results. Furthermore, the time required to equilibrate the trajectories is non-

negligible, particularly for computationally expensive approaches such as FPMD. Therefore, the reduction of the CC even in the trajectory generation step is of great importance.

Theoretically, a myriad of potential approaches can be used to generate independent trajectories. Here, we compare two potential schemes as examples to determine their effectiveness. The first scheme (scheme 1) generates each individual trajectory at a different random position in phase space, using random numbers to seed initial atomic velocities and random initial displacements from the initial equilibrium atomic positions. As an alternative for liquids or disordered systems, one could use random displacements from an initial seed configuration. For solids, one could enforce a maximum magnitude for the random displacements equal to the average amplitude of vibrations at the desired simulation temperature (i.e., $\sqrt{2k_BT/EL_c}$, where k_B, T, E, and L_c represent the Boltzmann constant, the temperature of the simulation, elastic modulus, and the characteristic length of the system under study, respectively). Utilizing larger amplitudes may have adverse effects and cause instabilities in the system.[27,28] After generating randomly positioned trajectories throughout phase space, we use velocity rescaling to equilibrate the individual trajectories at the temperature at which properties need to be calculated (i.e., the target temperature). The illustration of scheme 1 can be seen in Fig. 1(a).

To reduce the CC of generating the trajectories, we developed a second scheme (scheme 2) that unlike scheme 1, generates all the trajectories from a common initial point in phase space and therefore can reduce the CC associated with trajectory generation by as much as a factor of two. To make the N trajectories more distant in phase space as quickly as possible, scheme 2 includes three stages with different temperatures assigned to each stage. To implement the chosen temperatures, we use velocity rescaling throughout all of the stages. In addition, to further increase the distance between the trajectories in phase space, the frequency at which the velocities are rescaled are chosen randomly for every single simulation in scheme 2. Velocity rescaling is a simple yet effective algorithm for spreading trajectories in phase space.

To move a single equilibrated ensemble from one point in phase space to another, we can randomly perturb the positions and velocities of the atoms. If a microcanonical ensemble is used (NVE), this approach, however, would likely lead to a different temperature. This is unacceptable for many situations because many properties vary with the temperature, and very often we are interested in the average value of a property at a specific temperature. Therefore, it is necessary to control the temperature of the perturbed trajectory, which can be achieved by rescaling the velocities of the initial trajectory to the values corresponding to the desired system temperature. In addition, within the range of temperature fluctuations, the trajectory can explore different surfaces of constant energy in phase space according to $\Delta E = 3/2Nk_B(\sigma T)$. Here, ΔE represents the range of energy fluctuations, N is the number of atoms in the system, k_B is the Boltzmann constant, and σT is the standard deviation of the simulation temperature. In this view, velocity rescaling is a useful tool for generating new trajectories. Its application on any MD time step perturbs the system to a new position in phase space that is different than the preceding step, yet it does not drastically change the temperature or the energy of the simulation and therefore most often leads to the same macrostate.

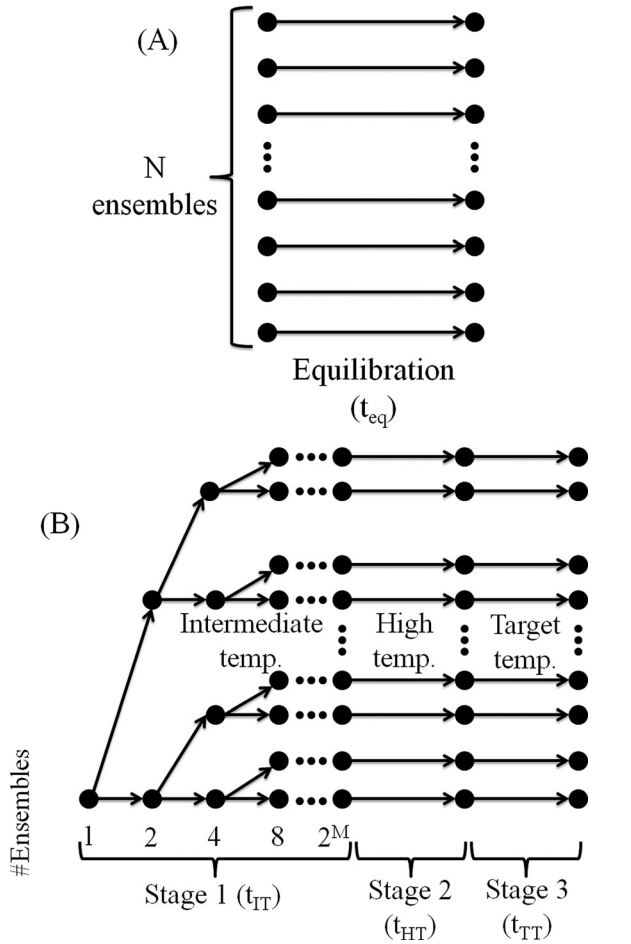

FIG. 1. Two schemes for trajectory generation. (a) Scheme 1: N trajectories are generated randomly in phase space via random displacements and velocities from an initial seed configuration, such as the equilibrium lattice sites for solids. After an initial equilibration period, phase space sampling is executed to gather data about the property of interest. (b) Scheme 2: The number of trajectories is doubled at each branching step in stage 1 until the desired number of trajectories (2^M) are generated at step M. Simulation temperatures are increased to a high temperature in stage 2 and then are decreased to the target temperature in stage 3. The simulation time lengths for stages 1–3 are t_{IT}, t_{HT}, and t_{TT}, respectively.

In stage 1 of scheme 2 of trajectory generation, the objective is to increase the number of trajectories from one to the desired value of N via branching from the initial point. We start with a single trajectory at an initial point and simulate it for a number of MD time steps using velocity rescaling. Then, we create a second trajectory from the first trajectory by randomly perturbing the velocities of the first trajectory to a different temperature. We keep the original NVE ensemble for the first trajectory, and now the second trajectory is set onto a different path since it is the product of a single NVT step at a different temperature. This process in essence branches one trajectory into two and the process can be repeated at various points to quickly generate a large number of trajectories. For example, by following the same

strategy, where the number of trajectories is doubled at each branching step, and if M branching steps occur during in stage 1 (e.g., the trajectory generation stage), then 2^M trajectories will be generated by the end of stage 1. During stage 1, the temperature for every single simulation is chosen randomly between the target temperature and a high temperature (i.e., a temperature larger than the target temperature). The high temperature should be assigned in such a way so that it does not induce any unwanted structural changes to the system under study (e.g., phase transitions or atomic dislocations). We refer to these random temperatures for the simulations in stage 1 as intermediate temperatures (i.e., between the target and the high temperatures).

In stages 2 and 3 of scheme 2 of trajectory generation, the objective is, first, to uniformly disperse the trajectories that are previously generated in stage 1 throughout phase space, and second, to have equilibrated trajectories at the end of scheme 2. To achieve this goal, in stage 2, we raise the temperature for all the trajectories to a high temperature (i.e., the same high temperature considered for stage 1). Then, in stage 3, we reduce the temperatures of the trajectories to the target temperature, so that the trajectories are now in equilibrium at the target temperature and are ready for phase space sampling. The schematic for scheme 2 can be seen in Fig. 1(b). The length of simulation times for stages 1–3 are denoted by intermediate temperature time (t_{IT}), high temperature time (t_{HT}), and target temperature time (t_{TT}), respectively. The length of these simulation times devoted for ensemble production schemes depends on the specific property and the system under study.

III. INDEPENDENCE PARAMETERS

In devising trajectory generation methods, we can use two metrics to quantify the relative instantaneous distance between two trajectories in phase space. We refer to these metrics as independence parameters (IP) for both position (IP_r) and velocity (IP_v) variables. These parameters are measures of orthogonality/similarity among the trajectories[29] and are defined as

$$IP_{\mathrm{r}} = \frac{\sum_{i,i'} \mathbf{r}_i \cdot \mathbf{r}_{i'}}{\left(\left[\sum_{i} \mathbf{r}_i \cdot \mathbf{r}_i\right]\left[\sum_{i'} \mathbf{r}_{i'} \cdot \mathbf{r}_{i'}\right]\right)^{1/2}}, \quad (1)$$

$$IP_{\mathrm{v}} = \frac{\sum_{i,i'} \mathbf{v}_i \cdot \mathbf{v}_{i'}}{\left(\left[\sum_{i} \mathbf{v}_i \cdot \mathbf{v}_i\right]\left[\sum_{i'} \mathbf{v}_{i'} \cdot \mathbf{v}_{i'}\right]\right)^{1/2}}, \quad (2)$$

where IP_{r} and IP_{v} are evaluated for two distinct trajectories in phase space using the atomic positions ($\mathbf{r}$) and velocities ($\mathbf{v}$), where atoms in one trajectory are denoted by the subscript i and atoms in the second trajectory are denoted by subscript i'. Since positions and velocities are functions of time, IP_{r} and IP_{v} are also functions of time, and the normalization criterion in the denominator ensures that IP_{r} and IP_{v} vary between zero and one. IP_{r} or IP_{v} equal one when two trajectories are located exactly at the same point in phase space. As they become more and more distant in phase space, IP_{r} and IP_{v} will decrease and approach zero, which corresponds to the case of completely independent trajectories. The variation of the independence parameters during schemes 1 and 2 of trajectory generation in the solid argon simulation is shown in Fig. 2. Figure 2 shows that both IP_{r} and IP_{v} are small for scheme 1, suggesting that they are spread uniformly in phase space, which can lead to efficient ensemble averaging. This was expected since the trajectories were chosen randomly in phase space, which guarantees their uniform distribution to a high degree. Furthermore, the combination of the high and intermediate temperatures throughout the three stages of scheme 2 allows the initial trajectories to separate from each other in phase space much more quickly (faster decrease of IP_{r} and IP_{v} (Fig. 3, top panel)) than if we used a constant target temperature throughout the stages (slower decrease of IP_{r} and IP_{v} (Fig. 3, bottom panel)).

IV. CASE STUDY 1: SOLID ARGON; RESULTS AND DISCUSSION

To compare the CC of the time averaging and the ensemble averaging methods and to evaluate the two trajectory generation schemes, we calculate the thermal conductivities of solid argon and silicon structures. For this purpose, we utilize equilibrium MD simulations and Green-Kubo formalism.[30,31] Based on Green-Kubo expression, thermal conductivity is proportional to the integral of the heat flux autocorrelation function (HFACF). In case of ensemble averaging, the HFACF is obtained by averaging over the

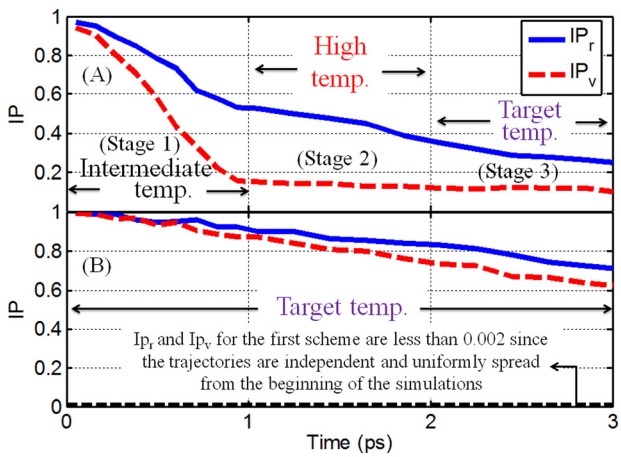

FIG. 2. Variation of independence parameters (IP) for schemes 1 and 2 for trajectory generation. Utilizing different temperatures at each stage of scheme 2 (a) will cause faster decrease in the IP_{r} and IP_{v} than the case of utilizing the same target temperature for all of the stages (b). Increasing the simulation time lengths for stages 1–3 decreases the IP_{r} and IP_{v} values to the point that they become closer to zero (i.e., completely independent trajectories) at the end of scheme 2 for trajectory generation. However, in our system of study, we confirmed that IP_{r} and IP_{v} values less than about 0.3 were sufficient to efficiently sample the phase space via the available ensembles (i.e., the calculated value for thermal conductivity changed by less than 1% for IP_{r} and IP_{v} values < 0.3). (b) Also shows the values of IP_{r} and IP_{v} for scheme 1 (black dashed line), which are small from the moment that the trajectories were chosen in phase space.

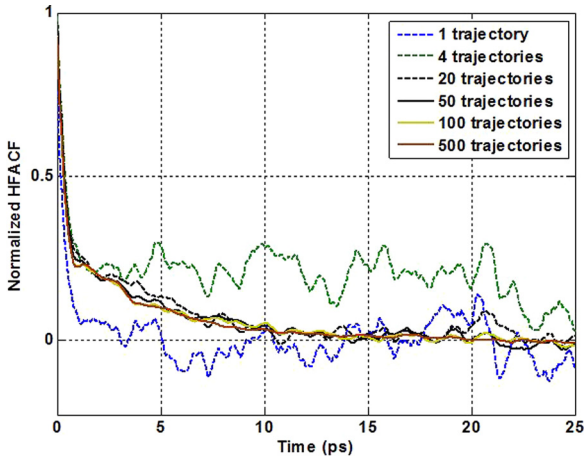

FIG. 3. Effect of increasing the number of trajectories on the convergence of the HFACF. By increasing the number of trajectories in the ensemble averaging method, the HFACF decays and converges. Each trajectory is simulated for 50 ps phase space sampling, and a 25 ps correlation length is calculated for the HFACF.

individually calculated HFACFs from independent trajectories. Except in special cases,[32] the HFACF decays to zero, causing the integral to converge. Therefore, in the calculations of thermal conductivity, a simulation length equal to at least the decay time of the HFACF is necessary. The integral of the HFACF can be calculated periodically during an MD simulation (i.e., by post-processing), which allows the user to terminate the simulations upon convergence. Increasing the number of parallel processors involved in time averaging increases the rate of time averaging to the point where the computation is dominated by only the communication time between processors. Conversely, in ensemble averaging, if sufficient computational resources (i.e., number of processors) are available, we can evaluate all of the trajectories concurrently. As a result, while ensemble averaging can scale linearly and indefinitely with the number of processors, time averaging becomes saturated when the communication time becomes rate limiting.

In the following examples, we performed all MD simulations using the Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS) package.[33] We obtained the instantaneous heat flux in the simulation according to the quantum energy flux operator derived by Hardy.[30,34] For solid argon, we assigned the simulation temperature to be 40 K. Furthermore, we used the Lennard-Jones potential,[9,35] a time step of 1 fs, and a face-centered-cubic structure that consists of 500 atoms in a $5 \times 5 \times 5$ unit-cell simulation box with periodic boundary conditions applied in all three dimensions. For solid argon, in scheme 1 of trajectory generation, we equilibrated the randomly chosen ensembles in phase space for 3 ps (t_{eq} in Fig. 1(a)). For scheme 2, the simulation time length for each of the three stages was considered to be equal to 1 ps, which results in the same total of 3 ps simulation time for scheme 2 (i.e., $t_{IT} + t_{HT} + t_{TT}$) as scheme 1. In addition, we chose the high temperature in scheme 2 of trajectory generation to be equal 80 K. We generated 512 trajectories using both of the trajectory generation schemes.

The main difference between calculating the thermal conductivity of solid argon by time averaging and ensemble averaging is the time devoted to the data acquisition step (i.e., phase space sampling after the generation of ensembles), during which instantaneous heat flux is recorded. The data acquisition step in time averaging was about 1 ns.[36] However, in ensemble averaging, we recorded the instantaneous heat flux in each trajectory for only 50 ps, which is twenty times shorter than the 1 ns of data acquisition time in time averaging. If we increase the number of trajectories from one to 500 and at each point calculate the HFACF by averaging it over these trajectories, the obtained HFACF becomes gradually smoother until the point of convergence (Fig. 3). By integrating the HFACF from 0 to 25 ps (it should be noted that other techniques for calculating thermal conductivity from Green-Kubo method, such as first avalanche[37] approach are also compatible with all the techniques presented herein), we calculated the thermal conductivity of solid argon to be 0.28 ± 0.02 W/(m K), which is in reasonable agreement with other reported values (e.g., 0.21 W/(m K)) by Kaburaki *et al.*[36] and 0.29 W/(m K) by Tretiakov and Scandolo.[38] The number of trajectories considered in ensemble averaging mainly depends on the parallel processing resources at hand and the physical background of the property under investigation. For instance, some transport properties may require larger numbers of trajectories for their convergence.[19]

Figure 4 shows the CC for the convergence of thermal conductivity calculations for both time averaging and ensemble averaging. In time averaging, increasing the number of MD simulation time steps increases the CC. On the other hand, in ensemble averaging, the CC is proportional to the number of trajectories included (i.e., n) in the calculations.

Another issue arises in how one should combine the results of a given number of trajectories to obtain the results. For example, if one has data available from N trajectories and wants to assess how the result varies for $n < N$

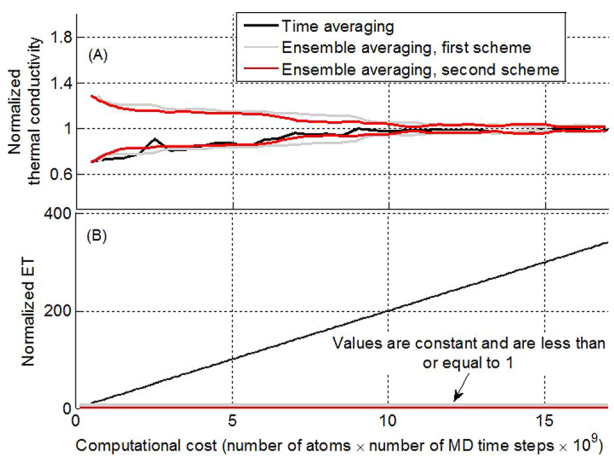

FIG. 4. The comparison of CC between time averaging and the two schemes for trajectory generation in calculation of thermal conductivity for solid argon. (a) By increasing the CC, the thermal conductivities calculated by time averaging and ensemble averaging converge to the same value. The values for thermal conductivity are normalized by the value from the time averaging. (b) For time averaging, the ET increases linearly and indefinitely with the CC. However, for ensemble averaging, the ET is a constant value. The values of the ET are normalized by the ET for the simulation of one trajectory using scheme 1 in ensemble averaging method.

trajectories, to determine the slope of convergence at N, one has many choices for how to pick the n trajectories out of the total N. Different choices of individual trajectories for the same value n, would subsequently lead to drastically different results. For example, consider $n = 1$, where the user has N possible choices. The variability of the results for a given single trajectory would be large and one might conclude different convergence behavior for different choices of n trajectories. For this reason, the standard deviations of the calculated values for thermal conductivity at each value of n (e.g., different values of CC) are chosen as the representative metric for how ensemble averaging converges to the final value of thermal conductivity. In this way, the standard deviations vary as two symmetric lines around the final converged value of thermal conductivity, which is calculated by inclusion of the total number of trajectories (N). The magnitude of the standard deviation decreases by increasing n, which directly indicates convergence for the ensemble averaging method. Figure 4(a) shows that both time averaging and ensemble averaging methods converge to the same value of thermal conductivity, which confirms that the system is ergodic. Furthermore, using ensemble averaging, the CC required to have a converged value of argon thermal conductivity was calculated to be 1.5 times larger than that of time averaging (Fig. 4(a)). The comparison of ET is also presented in Fig. 4(b). It can be seen that continued phase space sampling increases the ET for time averaging linearly and indefinitely, but not for ensemble averaging, if all of the trajectories are simulated concurrently. In this manner, for the calculations of argon thermal conductivity, the ET in ensemble averaging was determined to be 200 times less than the ET in time averaging.

V. CASE STUDY 2: CRYSTALLINE SILICON; RESULTS AND DISCUSSION

For solid argon, using ensemble averaging, the CC to have a converged value of thermal conductivity was calculated to be 1.5 times larger than that of time averaging (Fig. 4(a)). To make sure that our conclusion is independent of the specific material and inter-atomic potential considered, we also calculated the thermal conductivity of crystalline silicon, using Tersoff potential.[39] Our structure consists of 1000 atoms, the temperature for the simulation is chosen to be equal to 300 K, and a time step of 1 fs is considered for the MD simulation. For silicon, in scheme 1 of trajectory generation, we equilibrated the randomly chosen trajectories in phase space for 12 ps (t_{eq} in Fig. 1(a)). For scheme 2 of trajectory generation, the simulation time length for each of the three stages was considered to be equal to 4 ps, which results in the same total of 12 ps simulation time for scheme 2 (i.e., $t_{IT} + t_{HT} + t_{TT}$) as scheme 1. In addition, we chose the high temperature in scheme 2 of trajectory generation to be equal 400 K. We generated 512 trajectories using both of the trajectory generation schemes. The effect of increasing the number of trajectories on the convergence of HFACF for silicon structure is shown in Fig. 5. By integrating the HFACF from 0 to 400 ps, we calculated the thermal conductivity of crystalline silicon to be 183 W/(m K). Considering

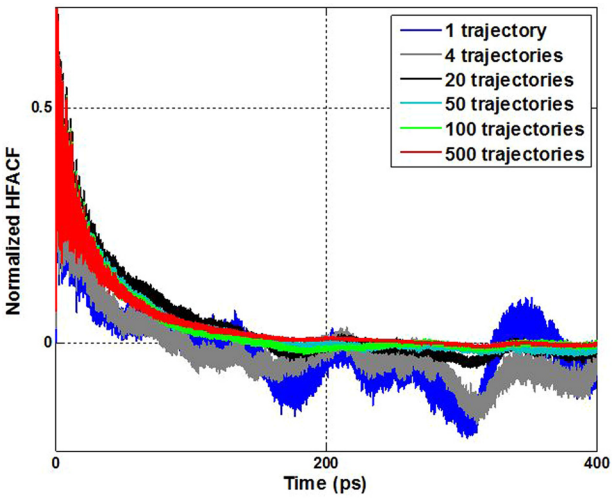

FIG. 5. Effect of increasing the number of trajectories on the convergence of the HFACF in silicon structure. By increasing the number of trajectories in the ensemble averaging method, the HFACF decays and converges. Each trajectory is simulated for 800 ps phase space sampling, and a 400 ps correlation length is calculated for the HFACF.

differences in inter-atomic potential and temperature, our calculated value is in reasonable agreement with the reported thermal conductivity values. For instance, value of 161 W/(m K) is reported for 300 K using EDIP inter-atomic potential[30] and value of 122 W/(m K) is reported for 500 K using Tersoff inter-atomic potential.[40] Figure 6 shows the comparison between time averaging and ensemble averaging for the calculation of the thermal conductivity of silicon. To have a converged value of silicon thermal conductivity, the CC of the ensemble averaging is 2.5 times larger than that of the time averaging. However, the ET for ensemble averaging is 10 times less than the time averaging, and it does not increase with increased CC if the trajectories are simulated

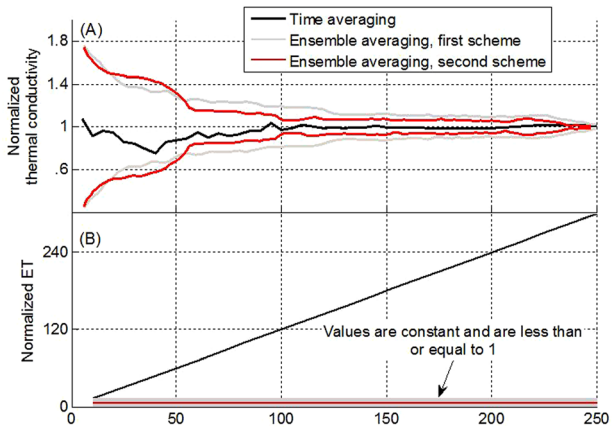

FIG. 6. The comparison of CC between time averaging and the two schemes for trajectory generation in calculation of thermal conductivity for crystalline silicon. (a) By increasing the CC, the thermal conductivities calculated by time averaging and ensemble averaging converge to the same value. The values for thermal conductivity are normalized by the obtained value from the time averaging. (b) For time averaging, the ET increases linearly and indefinitely with the CC. However, for ensemble averaging, the ET is a constant value. The values of the ET are normalized by the ET for the simulation of one trajectory using scheme 1 in ensemble averaging method.

concurrently. The agreement between the time sampled and ensemble averaged results again confirms the ergodic behavior of the silicon system.

VI. CONCLUSION

In this report, we calculated the thermal conductivities of solid argon and silicon structures using time averaging and ensemble averaging methods. The ET in time averaging is proportional to the amount of phase space data that is needed to be collected from the simulations. However, the same amount of phase space data can be collected from independent trajectories, and if sufficient parallel processing resources are available for the concurrent simulation of all the trajectories, ensemble averaging can decrease the ET significantly compared to time averaging. In addition, our calculations showed that the CC for both time averaging and ensemble averaging approaches are similar for thermal conductivity calculations. Simulating solid argon and silicon structures with two distinct inter-atomic potentials indicates that our conclusions are independent of the material and inter-atomic potentials. Furthermore, we presented two different schemes for the generation of uniformly spaced trajectories in phase space. In contrast to scheme 1, which is the direct random generation of trajectories throughout phase space, scheme 2 generates all the trajectories from a common initial point in phase space using velocity rescaling. It was shown that velocity rescaling is an efficient method for dispersing the trajectories quickly and uniformly throughout phase space. Although our calculations were based on classical MD, we believe that utilizing ensemble averaging methods with scheme 2 can also expedite the retrieval of the results of more expensive simulations, such as FPMD.

ACKNOWLEDGMENTS

Work at ORNL was supported by the Department of Energy, BES, through the S3TEC Energy Frontier Research Center.

[1]J. B. Clarage *et al.*, "A sampling problem in molecular dynamics simulations of macromolecules," Proc. Natl. Acad. Sci. **92**(8), 3288–3292 (1995).
[2]M. P. Allen and D. J. Tildesley, *Computer Simulation of Liquids* (Oxford University Press, 1989).
[3]K. Hwang, J. Dongarra, and G. C. Fox, *Distributed and Cloud Computing: From Parallel Processing to the Internet of Things* (Morgan Kaufmann, 2013).
[4]T. L. Hill, *Statistical Mechanics: Principles and Selected Applications* (Courier Dover Publications, 2013).
[5]S. M. Larson *et al.*, "Thoroughly sampling sequence space: Large-scale protein design of structural ensembles," Protein Sci. **11**(12), 2804–2813 (2002).
[6]U. H. Hansmann and Y. Okamoto, "Numerical comparisons of three recently proposed algorithms in the protein folding problem," J. Comput. Chem. **18**(7), 920 (2015).
[7]A. Mitsutake, Y. Sugita, and Y. Okamoto, "Generalized ensemble algorithms for molecular simulations of biopolymers," Pept. Sci. **60**(2), 96–123 (2001).
[8]M. F. Shlesinger, "Statistical mechanics: Exploring phase space," Nature **405**(6783), 135–137 (2000).
[9]A. McGaughey and M. Kaviany, "Thermal conductivity decomposition and analysis using molecular dynamics simulations. Part I. Lennard-Jones argon," Int. J. Heat Mass Transfer **47**(8), 1783–1798 (2004).
[10]S. G. Volz and G. Chen, "Molecular-dynamics simulation of thermal conductivity of silicon crystals," Phys. Rev. B **61**(4), 2651 (2000).
[11]J. Yang, Y. Wang, and Y. Chen, "GPU accelerated molecular dynamics simulation of thermal conductivities," J. Comput. Phys. **221**(2), 799–804 (2007).
[12]S. G. Volz and G. Chen, "Molecular dynamics simulation of thermal conductivity of silicon nanowires," Appl. Phys. Lett. **75**(14), 2056–2058 (1999).
[13]J. R. Lukes and H. Zhong, "Thermal conductivity of individual single-wall carbon nanotubes," J. Heat Transfer **129**(6), 705–716 (2007).
[14]Y. Chen *et al.*, "Molecular dynamics study of the lattice thermal conductivity of Kr/Ar superlattice nanowires," Phys. B: Condens. Matter **349**(1), 270–280 (2004).
[15]J. R. Lukes and C. Tien, "Molecular dynamics simulation of thermal conduction in nanoporous thin films," Microscale Thermophys. Eng. **8**(4), 341–359 (2004).
[16]A. McGaughey and M. Kaviany, "Thermal conductivity decomposition and analysis using molecular dynamics simulations. Part II. Complex silica structures," Int. J. Heat Mass Transfer **47**(8), 1799–1816 (2004).
[17]K. Gordiz and S. M. V. Allaei, "Thermal rectification in pristine-hydrogenated carbon nanotube junction: A molecular dynamics study," J. Appl. Phys. **115**(16), 163512 (2014).
[18]B. Mortazavi and Y. Rémond, "Investigation of tensile response and thermal conductivity of boron-nitride nanosheets using molecular dynamics simulations," Phys. E **44**(9), 1846–1852 (2012).
[19]Y. Chalopin *et al.*, "Thermal interface conductance in Si/Ge superlattices by equilibrium molecular dynamics," Phys. Rev. B **85**(19), 195302 (2012).
[20]N. de Koker, "Thermal conductivity of MgO periclase from equilibrium first principles molecular dynamics," Phys. Rev. Lett. **103**(12), 125902 (2009).
[21]J. Che *et al.*, "Thermal conductivity of diamond and related materials from molecular dynamics simulations," J. Chem. Phys. **113**(16), 6888–6900 (2000).
[22]P. Lindan *et al.*, "First-principles molecular dynamics simulation of water dissociation on TiO_2 (110)," Chem. Phys. Lett. **261**(3), 246–252 (1996).
[23]K. Ohno *et al.*, "*Ab initio* Molecular dynamics simulations for collision between C 60- and alkali-metal ions: A possibility of Li@ C 60," Phys. Rev. Lett. **76**(19), 3590 (1996).
[24]H. Zhao and N. Aluru, "Molecular dynamics simulation of bulk silicon under strain," Interact. Multi. Mech. **1**(2), 303–315 (2008).
[25]J. S. Tse, "*Ab initio* molecular dynamics with density functional theory," Annu. Rev. Phys. Chem. **53**(1), 249–290 (2002).
[26]Z. Fan, T. Siro, and A. Harju, "Accelerated molecular dynamics force evaluation on graphics processing units for thermal conductivity calculations," Comput. Phys. Commun. **184**(5), 1414–1425 (2013).
[27]L. Cartz, "Thermal vibrations of atoms in cubic crystals II: The amplitude of atomic vibrations," Proc. Phys. Soc. London, Sect. B **68**(11), 957 (1955).
[28]E. G. Gamaly, *Femtosecond Laser-Matter Interaction: Theory, Experiments and Applications* (CRC Press, 2011).
[29]C. H. Bishop and G. Abramowitz, "Climate model dependence and the replicate Earth paradigm," Clim. Dyn. **41**(3–4), 885–900 (2013).
[30]A. S. Henry and G. Chen, "Spectral phonon transport properties of silicon based on molecular dynamics simulations and lattice dynamics," J. Comput. Theor. Nanosci. **5**(2), 141–152 (2008).
[31]A. Henry and G. Chen, "Anomalous heat conduction in polyethylene chains: Theory and molecular dynamics simulations," Phys. Rev. B **79**(14), 144305 (2009).
[32]A. Henry and G. Chen, "High thermal conductivity of single polyethylene chains using molecular dynamics simulations," Phys. Rev. Lett. **101**(23), 235502 (2008).
[33]S. Plimpton, "Fast parallel algorithms for short-range molecular dynamics," J. Comput. Phys. **117**(1), 1–19 (1995).
[34]R. J. Hardy, "Energy-flux operator for a lattice," Phys. Rev. **132**(1), 168 (1963).
[35]J. R. Lukes *et al.*, "Molecular dynamics study of solid thin-film thermal conductivity," J. Heat Transfer **122**(3), 536–543 (2000).
[36]H. Kaburaki, J. Li, and S. Yip, "Thermal conductivity of solid argon by classical molecular dynamics," MRS Proceedings **538**, 503 (1998).

[37]J. Chen, G. Zhang, and B. Li, "How to improve the accuracy of equilibrium molecular dynamics for computation of thermal conductivity?," Phys. Lett. A **374**(23), 2392–2396 (2010).
[38]K. V. Tretiakov and S. Scandolo, "Thermal conductivity of solid argon from molecular dynamics simulations," J. Chem. Phys. **120**(8), 3765–3769 (2004).
[39]J. Tersoff, "Modeling solid-state chemistry: Interatomic potentials for multicomponent systems," Phys. Rev. B **39**(8), 5566 (1989).
[40]C. A. da Cruz *et al*., "Molecular dynamics simulations for the prediction of thermal conductivity of bulk silicon and silicon nanowires: Influence of interatomic potentials and boundary conditions," J. Appl. Phys. **110**(3), 034309 (2011).