A comparison of the efficiency of Fourier- and discrete time-path integral Monte Carlo

C. Chakravarty
Department of Chemistry, Indian Institute of Technology, Hauz Khas, New Delhi 110016, India

M. C. Gordillo and D. M. Ceperley[a)]
Department of Physics, University of Illinois, 1110 West Green Street, Urbana, Illinois 61801

(Received 3 November 1997; accepted 24 April 1998)

We compare the efficiency of Fourier and discrete time path integral Monte Carlo (PIMC) methods on a cluster of 22 hydrogen molecules at 6 K. The discrete time PIMC with a pair density matrix approximation to the path action is shown to be the most efficient for evaluating all the observables studied here. The Fourier PIMC technique has a comparable efficiency for observables diagonal in the coordinates but is significantly worse for estimating the kinetic and total energies. The superior performance of the discrete time PIMC is shown to be due to the more accurate treatment of the path action using the pair density matrix approach; the discrete time PIMC simulation within the primitive approximation is much less efficient. Complete details of the implementation of all algorithms are given. © *1998 American Institute of Physics.* [S0021-9606(98)50829-X]

I. INTRODUCTION

Path integral methods are an important simulation tool, especially for the treatment of strongly interacting quantum systems. In the imaginary-time path integral method one maps a quantum model onto a ''classical system'' of *paths* to which we can apply classical simulation techniques and obtain the observables of interest. Using path integral simulations one can, in practice, accurately estimate thermodynamic properties of many-body systems all the way from the classical high-temperature regime to temperatures approaching the ground state.[1]

The expression to obtain the expectation value of an operator O at temperature T is, in coordinate representation

$$\langle O\rangle = \frac{\int dR\ dR'\ \rho(R,R';\beta)\langle R|O|R'\rangle}{Z}, \tag{1}$$

where

$$Z = \int dR \rho(R,R;\beta) \tag{2}$$

is the partition function, $\beta = 1/k_B T$, and $\rho(R,R';\beta)$ is the many-body thermal density matrix. Here R and R' represent the position coordinates of all N particles. The density matrix is written as a path integral in imaginary time using the Feynman–Kac formula:[2]

$$\rho(R,R',\beta) = \oint D[R(u)]\ \exp(-S[R(u)]). \tag{3}$$

Here, u is the imaginary-time coordinate (always in units of $\hbar$) and $\oint D[R(u)]$ represents the integral over the set of all paths starting at R (time 0) and ending at configuration R' (time β). $S[R(u)]$ is the action of the path, given by

$$S[R(u)] = \int_0^\beta du\ \frac{m}{2\hbar^2}\left[\frac{dR}{du}\right]^2 + V[R(u)], \tag{4}$$

where m is the mass of a particle, and $V[R]$ is the potential energy at the configuration R.

In this paper, we compare two advanced related numerical implementations to evaluate the expectation values: the Fourier (F) and the discrete-time (DT) path integral Monte Carlo (PIMC) methods. Both methods have been used extensively to compute properties of important quantum systems. The methods differ in how the imaginary time paths are represented. In the first method, the Fourier components of the imaginary time trajectory are used to represent the path, while in the second method, the continuous path in imaginary time is represented in real space by a fixed number of points, usually called ''time-slices.'' Within the framework of the DT-PIMC, several approaches are possible depending on the approximation to the action. In this paper, we consider two approximate actions: the primitive and the pair actions.

Even though very significant computational resources have been consumed using these algorithms, a detailed comparison of their accuracy and computer time requirements has not been made, presumably because of the difficulty of coding and testing the corresponding algorithms. There are various reasons which motivate the choice of a particular algorithm including the availability of software, familiarity of the formalism, ease of programming and debugging, reliability and, finally, efficiency. In fact, although the first reasons are quite important in choosing a method, we will only compare their efficiencies. The other reasons are hard to quantify and, in the long term, less important if a general purpose implementation of the program is made publicly available.

However, it is not completely trivial to define what is actually meant by a more efficient method. Our general procedure is the following. First, one has to select a problem on

a)Electronic mail: ceperley@uiuc.edu

which to make the comparison. Some methods might be more suitable for doing certain problems than others. Second, one has to decide which properties are important and how accurately one needs to compute them: in other words, what is the target range for the total error. With these fixed, the relative efficiency of various methods is the ratio of computer time needed to attain those errors.

In path integral calculations, as in other simulation techniques, there can be both systematic errors, δ, and statistical errors, ϵ. The most important systematic error in path integral calculations arises from approximating continuous imaginary-time paths with a representation in a discrete basis. To make the comparison between methods, we must first determine the value of the discretization variable so that the systematic error is equal to a fixed value δ, the target value, in all methods. Once that is done, one runs each simulation until the statistical error (obtained from the fluctuations in the average during the simulation) is ϵ. Then the efficiency of any simulation method is:[3]

$$\zeta = \frac{1}{\epsilon^2 t}, \tag{5}$$

where t is the computer time used. So defined, the efficiency is independent of t since $\epsilon \propto t^{-1/2}$. From ζ we can determine the relative time we need to obtain statistical errors below any particular threshold. Obviously, the calculated efficiency depends on the observable of interest.

Clearly, to estimate the overall efficiency of a simulation method we need to have good estimates of the statistical errors for several properties, and the computer time required. The main complication in computing the statistical error comes about because of correlation between successive paths in the process of generating samples of the path with the Monte Carlo methods appropriate for many-body systems. We estimate the errors by using two related methods:[4] computing the autocorrelation time of the property and blocking the averages. The latter is done by dividing all the measurements we have of a particular property into blocks of the same length, and finding the average within each particular block. If successive blocks are uncorrelated, one can estimate the statistical errors from the fluctuations of the block averages. By studying how the estimated error depends on the block length one can determine the effect of the autocorrelation on the final average. In all cases presented here, we observe that the autocorrelation functions decay to zero in a small fraction of the total number of total blocks, so unbiased error estimates are obtained.

With respect to estimating the required computer time, one has the difficulty that the computer time will depend on many details such as the computer vendor and model, the compiler, the programmer's skill and amount of optimization for a particular problem. To ease things, we used the same computer in all three series of calculations presented here. To ensure that the codes were equally optimized, our strategy was to take two mature codes (the different actions in DT-PIMC were done with the same code) and run them on the same problem, one for which both methods have been used in the past. No further optimization was done. Of course, each method is capable of significant improvements (we return to this point in the summary), but a fair test is to do the comparison for the computer programs *as they have been used in practice.* Even so, we cannot be sure that differences in efficiency of less than a factor of 2 are significant.

One of the objectives of this paper is to give enough practical details to allow the calculations to be reproduced and permit future comparisons with methods yet to be developed. We feel that a comparison, even if slightly flawed, may help novices use the methods and realize how important the efficiency issues are. We will not justify the algorithms here, that is done in the references.

II. THE PHYSICAL SYSTEM

In choosing the physical system on which to do the comparison, we need to find one with strong quantum effects. We have decided to compare the efficiency of the algorithms on a cluster of 22 hydrogen molecules at a temperature of 6 K. Both algorithms[5–7] have been used previously on this system and no special problems with convergence have been reported. Since our objective is to compare the relative efficiencies of the techniques, details of the interaction potential are not crucial; however, we wish the model to be sufficiently realistic that it can be used as a benchmark for more realistic models.

The energy to excite a H_2 molecule to the $J=1$ state is 170 K; to the $J=2$ state, 510 K. Hence at 6 K the overwhelming majority of molecules are in the $J=0$ state and, to a good approximation, the hydrogen molecules are spherical bosons interacting with an isotropic pair potential. We assume the pair interaction between molecules to be a Lennard-Jones 6–12 form[6] with $\epsilon_{\mathrm{LJ}} = 34.2$ K and $\sigma_{\mathrm{LJ}} = 2.96$ Å. With these two parameters, the Lennard-Jones potential is able to reproduce most of the features obtained with the more realistic Silvera–Goldman potential.[8] We do not cut off the potential at large distance, because that is not needed in a cluster.

Since a cluster in vacuum at any positive temperature is metastable with respect to evaporation, we used a confining potential (a spherical cavity with radius about 12 Å) to prevent any molecules from permanently leaving the cluster. Even though such events are rare they will occur in a long run. The confining potential has the form:[9]

$$V_c = \epsilon_{\mathrm{LJ}} \sum_{i=1}^{N} [|\mathbf{r_i} - \mathbf{R}|/(4\sigma_{\mathrm{LJ}})]^{20}, \tag{6}$$

where $\mathbf{R}$ is the center of mass of the cluster (defined at each value of imaginary time) and $\mathbf{r}_i$ is the position of particle i. ϵ_{LJ} and σ_{LJ} have the values given above for the H_2–H_2 interaction. The confining potential is steep but still analytic so that we can take the second derivatives needed for partial averaging (see Sec. IV). This spherical confining potential has little effect on the final results at the temperature at which we will do the comparison.

We used 22 molecules in the cluster, each of them with a mass of 2 atomic units, giving a value of $\lambda = \hbar^2/2m = 12.1272$ K Å^2. The cluster is large enough to have several layers but small enough so that extensive runs

can be used to check the convergence of the MC calculations. Since $N=22$ is not a magic number for H_2 clusters, fluctuations in properties are not expected to be either unusually small or large. However, a cluster with this number of molecules has an incomplete top layer which could lead to slower convergence of the simulation. Having several shells means that there is a density gradient so that the systematic discretization errors will be dominated by the inner high-density region. In contrast, the convergence of the statistical error is controlled by surface fluctuations.

There is a very large depression of the melting temperature of bulk hydrogen at the surface,[10] so we expect the top layers of the cluster to be relatively mobile, while those in the center will be localized. At a higher temperature the vapor pressure increases so that the effective ''evaporation rate'' could lead to interaction with the spherical cavity. The temperature was chosen to be 6 K so that the system would be still liquid and highly quantum mechanical. At this temperature, quantum exchange of molecules is unimportant.[5]

The basic quantities we will compare are the components of the energy: the kinetic, potential and total energies. The potential energy is about -48 K/molecule and the kinetic energy is about 30 K/molecule (compared to the classical value of 9 K/molecule). We fixed the admissible systematic error, δ, to be 0.25 K for all the three kinds of energy. This accuracy is reasonable: it is about 1% of the chemical potential and about equal to the boson exchange energy. Although this error is somewhat smaller than used on cluster calculations in the past, precise calculations are needed to make comparisons of highly quantum systems with experiment and to refine intermolecular potentials.

Reiterating the procedure to find the efficiency of each method: first we determine the number of time slices or Fourier components necessary to obtain $\delta\approx 0.25$ K and, using that number, we see how much computer time was needed to obtain a certain statistical error, ϵ. The ratio of computer time does not depend on ϵ.

We will present the results in three sets of figures: first the energies versus the discretization parameter for the various methods (Fig. 4), then the statistical efficiency versus discretization (Fig. 5) and, finally, the ''true efficiency,'' the curve of computer time (Fig. 6) needed to achieve a given total error ($\epsilon=\delta$) as a function of δ. Using these curves, one can determine which method is most efficient for any target error. This is done separately for each component of the internal energy. In addition to the energy, we compared the results for some structural properties: the density profile (the probability of finding an atom at a distance r from the center of mass) and the angular distribution (the probability distribution for the values of the smallest angle for every triplet of molecules in the cluster). The results are given in Figs. 1 and 2. The data obtained are similar in the three approaches used.

In the next two sections we describe the details and results of each method.

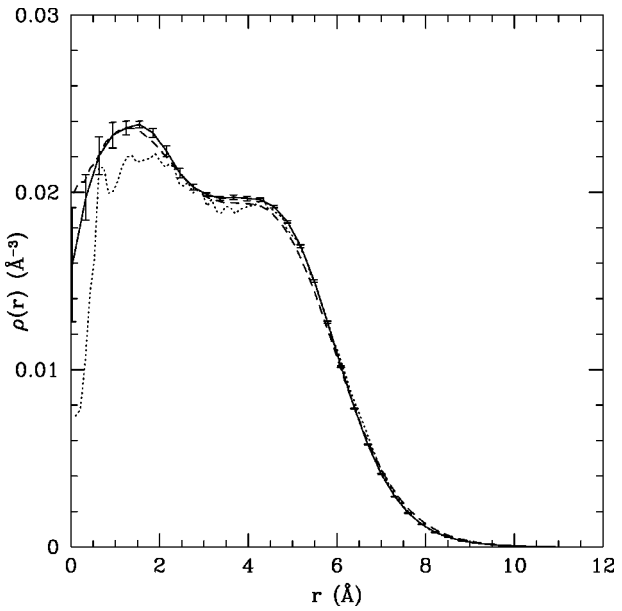


FIG. 1. The density profile as computed with the three methods. The solid line (with error bars) shows the result of the DT-PIMC using the pair action and 16 time slices; the dashed line DT-PIMC using the primitive approximation and 120 time slices; and the dotted line the F-PIMC with 36 Fourier components. The differences at the origin are due to statistical errors in sampling such a small volume.

III. DISCRETE TIME PATH INTEGRAL MONTE CARLO

In the discrete-time path integral Monte Carlo (DT-PIMC) method, one writes the partition function [Eq. (2)] as an integral over $(M-1)$ intermediate points or ''time-slices'' $\{R_2 \ldots R_M\}$:

$$Z=\int dR_1 \cdots dR_M \exp\left[-\sum_{i=1}^{M} S(R_i,R_{i+1};\tau)\right], \qquad (7)$$

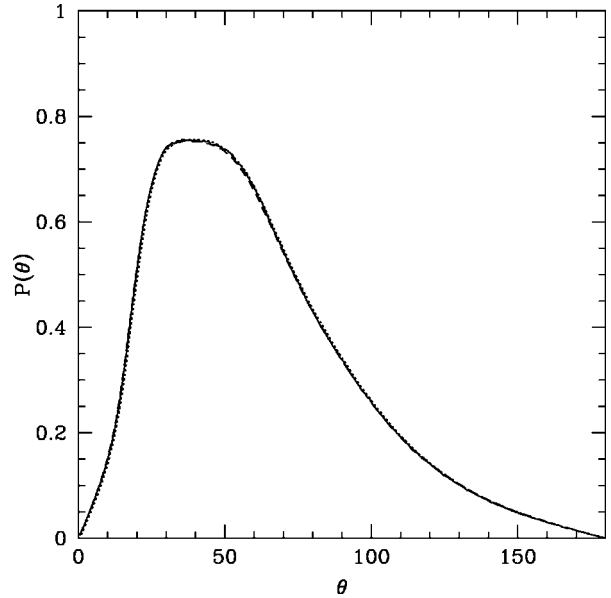


FIG. 2. The angular distribution as computed with the three methods. Shown is the probability distribution that a given triplet of molecules subtends an angle θ. The symbols and parameters are as in Fig. 1.

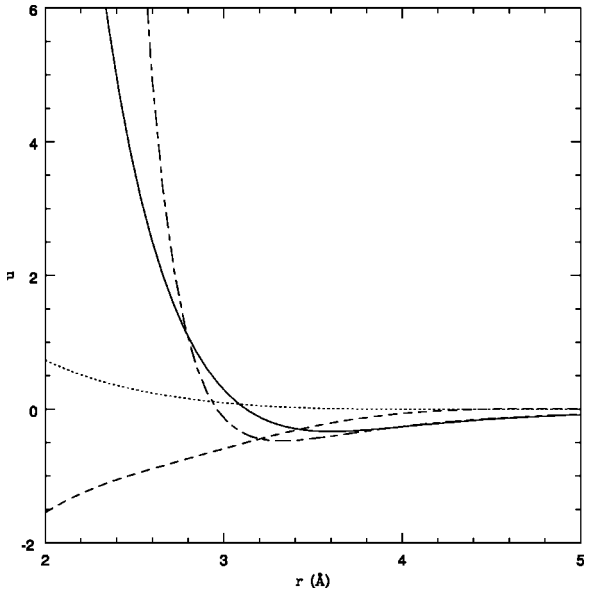


FIG. 3. The action components for the H_2-H_2 potential at a time step of $\tau=0.0139\ K^{-1}$. Short–long dashes, primitive approximation, $\tau v(r)$; solid line, the diagonal component, u_0; dotted curve, u_{10}; and dashed curve, u_{11}.

where R and S have the same meaning as before and $R_{M+1} \equiv R_1$. Of course, the action depends on the value of $\tau=\beta/M$. Then in DT-PIMC one makes an approximation to the exact action at a sufficiently small value of τ (large value of M) and evaluates the integrals with a Monte Carlo method.

There are three main ways in which a path integral algorithm can be optimized: by using more accurate forms for the action, by sampling more effectively and in ''estimating'' the energy with lower variance. Here we confine our remarks to a brief synopsis of the algorithm. A detailed description and justification is given in Ref. 2.

A. Action

It is important to choose an accurate action because if one can use a larger value of τ, one will have many fewer time slices and have a much more efficient algorithm. We split the action into the free-particle kinetic part and the rest which is due to the interaction:

$$S(R,R';\tau)=\frac{3N}{2}\ \ln\,(4\pi\lambda\tau)+\frac{(R-R')^2}{4\lambda\tau}+U(R,R';\tau). \tag{8}$$

The primitive action is given by

$$U_P(R,R';\tau)=\frac{\tau}{2}[V(R)+V(R')]. \tag{9}$$

Our second choice, a much more accurate action, is the pair action:[2,11]

$$U_2(R,R';\tau)=\sum_{i<j} u_2(\mathbf{r}_{ij},\mathbf{r}'_{ij};\tau), \tag{10}$$

where $u_2(\mathbf{r},\mathbf{r}';\tau)$ is the interacting part of the exact action for a pair of particles. For single component systems interacting with pair potentials, the exact pair action is a very accurate choice,[2] as the results below will confirm. This is determined numerically, before the simulation begins. To compute the exact action and energy for a pair of hydrogen molecules, we used a matrix squaring technique.[12] To handle the angular variable in the pair action we expand in a power series in the difference between the old and new coordinates as is given in Eq. (4.46) of Ref. 2. We used a value of $n=1$ so that we used the three radial functions $u_0(r),u_{10}(r),u_{11}(r)$ for the action and a value of $n=2$ (which has an additional three radial functions) for the computation of the energy. We store these functions $u_{ij}(r)$ in a table for use during the simulation; the values are obtained by cubic interpolation from the grid values. We estimate that the numerical errors in evaluating the pair action are less than 0.01. Figure 3 shows the components of the H_2-H_2 pair action. Consult Ref. 2 (section 4) for further details about procedures for computing and representing the pair action and for additional considerations going into choice of the action.

B. Sampling

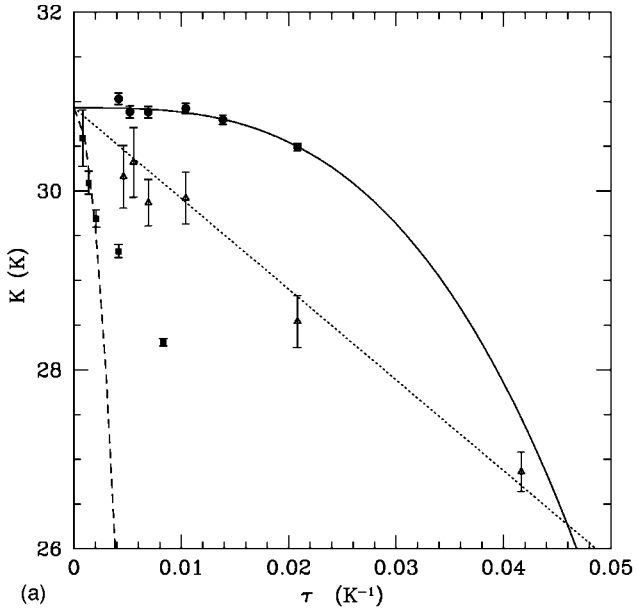
The sampling algorithm we have used is a generalization of the Metropolis[13] procedure: one initializes the full imaginary-time path in some fashion. Then one makes repeated attempts to change the path. A trial move, involving possibly several coordinates, is sampled around the old path. That move is either accepted or rejected. There are two basic types of moves we used. For this method we define a *pass* as consisting of one application of each procedure.

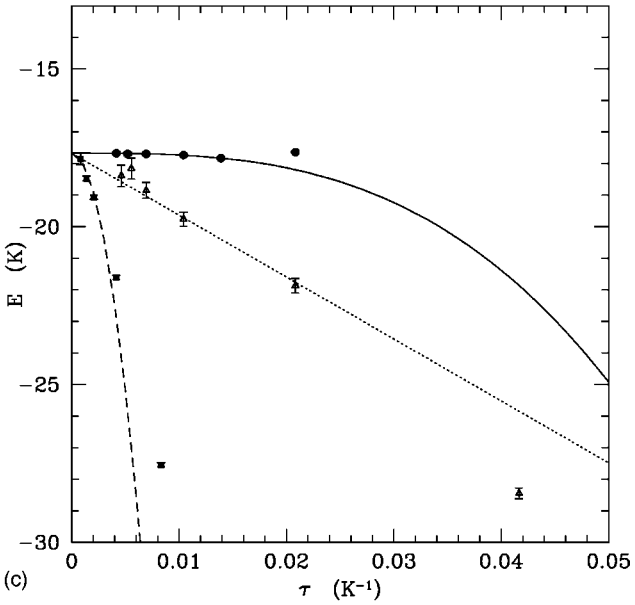
(1) Displacement move. All of the coordinates for a single molecule are displaced uniformly in a cube of size $\Delta_{\rm cm}$. This move is a generalization of what is often done classically to move a particle, except that all M time slices of a single particle are moved together. The move is accepted if the change in the action is less than minus the logarithm of a uniformly distributed random number. A displacement move does not change the ''trajectory'' of an individual particle in imaginary time. In addition, a displacement move takes longer than a bisection move (described next) because the action for all values of imaginary time need to be updated. In a single pass, an attempt was made to displace each molecule once. The acceptance ratio for a displacement move for $\Delta_{\rm cm}=0.5$ Å was 63%.

(2) Bisection moves. This is a recursive algorithm using multilevel sampling and the Levy construction.[2] During a single bisection move for a given molecule, we try to move 2^l-1 adjacent time slices. The parameter l is called the *level* of the sampling. We found that best results were obtained with level $l=3$ for $M>8$ and $l=2$ for $M=8$.

The procedure is as follows:

(a) Selection of molecule and time slices to be updated. The time slice ''origin'' was chosen at random in the range $(1-M)$. All the molecules are attempted in order. In the following, assume for simplicity that time slice origin is $\mathbf{r}_0$. This means that coordinates $\mathbf{r}_0,\mathbf{r}_8,\mathbf{r}_9,\ldots$ are held fixed and only $\mathbf{r}_1,\ \ldots,\ \mathbf{r}_7$ are attempted to be moved.





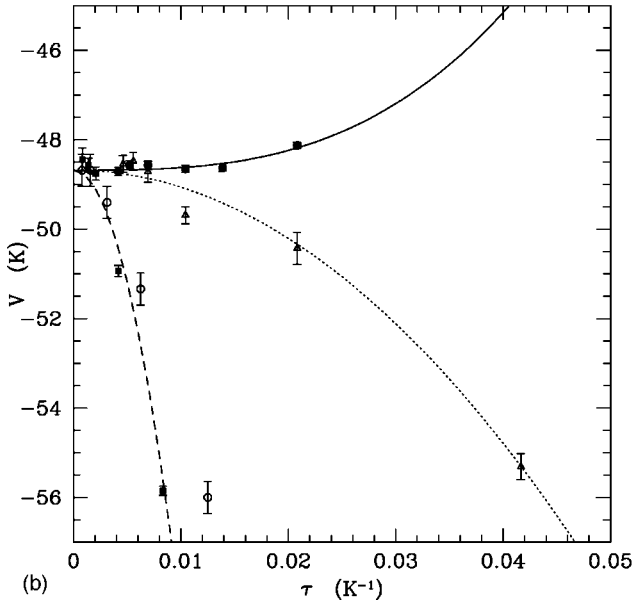


FIG. 4. Components of the energy as a function of the time step τ for DT-PIMC pair action (solid circles), primitive action (solid squares), and for F-PIMC (open triangles). Fits are shown by the curves. (a), (b), and (c) are the kinetic energy, potential energy, and total energy, respectively. In (b) is shown the scaled potential energy from Ref. 7, also using the primitive approximation and DT-PIMC (open circles).

(b) Bisection at top level ($l'=1$). The coordinate $\mathbf{r}_4$ is sampled using

$$\mathbf{r}_4' = \tfrac{1}{2}[\mathbf{r}_0 + \mathbf{r}_8] + \eta\sqrt{4\tau\lambda}, \tag{11}$$

where $\lambda = \hbar^2/2m$ and η is a normally distributed random three-vector with zero mean and unit variance. We then compute $U(R;4\tau)$, an approximation to the interacting part of the action for a path containing every fourth step and set

$$\delta U_{4\tau} = U(R';4\tau) - U(R;4\tau). \tag{12}$$

The trial position is accepted or rejected by comparing

$$A_{4\tau} = \min\,[1, \exp(-\delta U_{4\tau})] \tag{13}$$

with a uniformly distributed random number in [0,1]. The approximate action is only needed to provisionally accept new paths and does not affect the final answer. Thus one is free to make any convenient choice for it. We used the exact pair action evaluated at 4τ, except we did not include the off-diagonal terms in computing the pair action. If $\mathbf{r}_4'$ is rejected, one goes on to the next molecule.

(c) Bisect at the next level ($l'=2$). Coordinates $\mathbf{r}_2'$ and $\mathbf{r}_6'$ are then sampled using the above procedure and the previously sampled $\mathbf{r}_4'$ with the 4τ in Eqs. (11) and (12) becoming 2τ. The two trial midpoints are accepted or rejected together, based on

$$A_{2\tau} = \min\,[1, \exp(-\delta U_{2\tau} + \delta U_{4\tau})]. \tag{14}$$

(d) Bisect at the final level ($l'=3$). The coordinates $\mathbf{r}_1'$, $\mathbf{r}_3'$, $\mathbf{r}_5'$, and $\mathbf{r}_7'$ are sampled using the previously sampled points and accepted or rejected based on A_τ. Only if this step is accepted is the old path replaced by the new path.

The acceptance ratio at each level [steps (b)–(d) above] when the number of time slices $M=12$ is close to 50%. This makes the overall acceptance ratio for a multi-level move

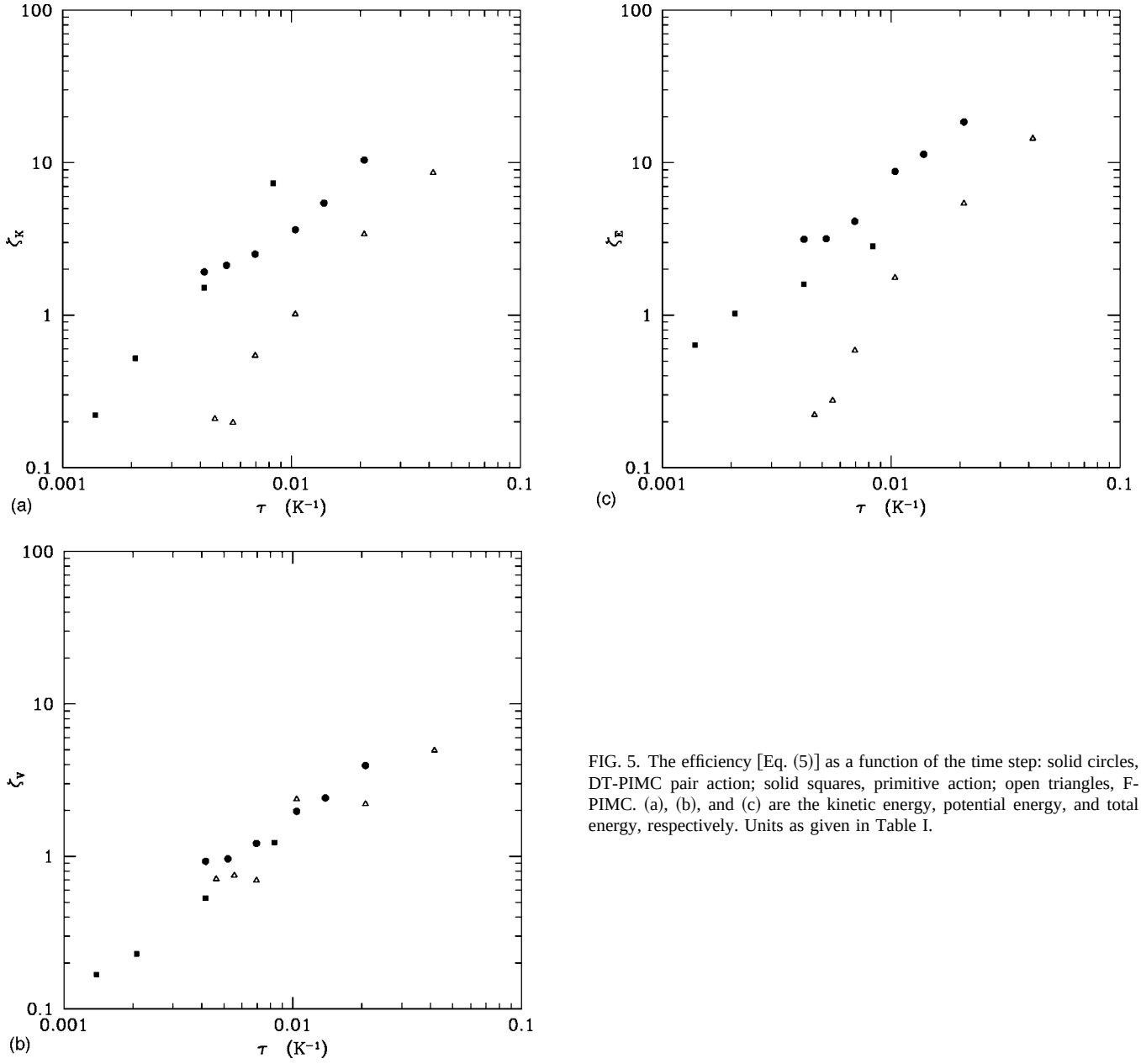


FIG. 5. The efficiency [Eq. (5)] as a function of the time step: solid circles, DT-PIMC pair action; solid squares, primitive action; open triangles, F-PIMC. (a), (b), and (c) are the kinetic energy, potential energy, and total energy, respectively. Units as given in Table I.

equal to 10% $\approx(0.5)^3$. If we decrease the value of τ, the acceptance ratio goes up. For example, at $M=40$ the overall acceptance ratio is about 40%.

The algorithmic parameters in DT-PIMC are the bisection level, the frequency of attempting displacement moves and their average size, Δ_{cm}. These were set by trying to approximately maximize the average movement through phase space. Possible elaborations of sampling, including correlated sampling, permutation and multi-particle moves, are needed for superfluids,[2] but not for this problem of hydrogen clusters at 6 K.

C. Computation of the energy

Since the energy is one of the primary results of a Monte Carlo calculation, special care needs to be taken to compute it with low variance. The simplest estimator is called the thermodynamic (T) estimator:

$$E=-\frac{dZ}{Zd\beta}. \tag{15}$$

However, the variance of this estimator increases as τ decreases. It can be shown,[14] that one can transform the thermodynamic estimator into a form with lower variance, the *virial estimator* of the energy. In our implementation, the virial estimator has the same systematic error due to the time step but a lower statistical error, particularly for nearly classical systems or small values of τ. For a derivation of the virial estimator, the reader is directed to Ref. 2. There is a free parameter in the virial estimator, the window size. We choose it to minimize the variance of kinetic energy by using a value of ≈ 0.04 K^{-1} for all the calculations shown in Table I. For $M=12$ the error bar of the virial estimator is 0.63 times the error of the thermodynamic (T or Barker) estimator (see Table I). This corresponds to an increase in efficiency of a factor of about 2.5 with very little increase in

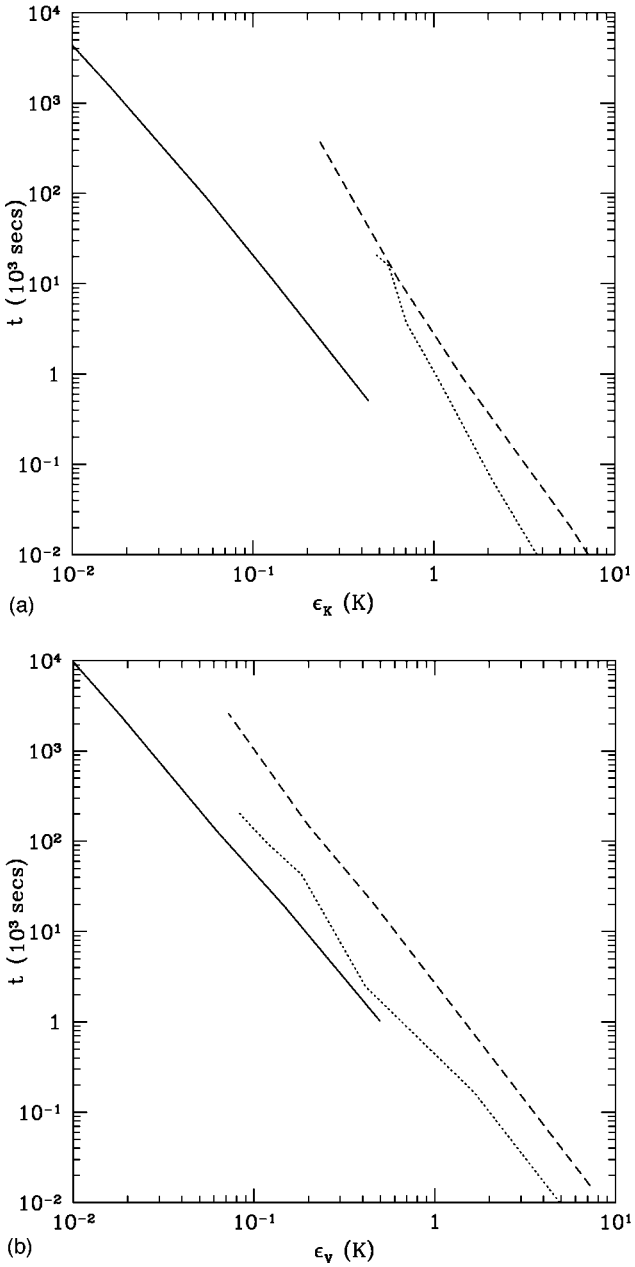


FIG. 6. The computer time needed to have both the statistical and systematic error less than ϵ as a function of ϵ. The solid line is the DT-PIMC with the pair action; dashed line, the DT-PIMC with the primitive action; and dotted line, F-PIMC. (a), (b), and (c) are the kinetic energy, potential energy, and total energy, respectively.

computer time. (It increases because one needs to compute the gradient of the action.) The energy is computed for the entire system and for all updated time slices once every pass.

We also added the polarization correction to the kinetic energy as given by Eq. (4.52) of Ref. 2. This contributes a negligible amount to the kinetic energy for $M \geqslant 12$ but provides a good estimate of the magnitude of neglected multiparticle contributions to the kinetic energy.

D. Results

Figure 4 and Tables I and II show the dependence of the kinetic, potential and total energies on τ for both the pair action and the primitive approximations. To ensure a good comparison, the algorithmic parameters (e.g., the number of levels, l) were kept identical.

Using the pair action, the figures show that for $\tau \leqslant 0.014$ K^{-1}, the values of each of the three energies are constant within 0.25 K. In order to have the systematic error $\delta <$ 0.25 K requires between 8 and 12 time slices with the pair action. One also notices from the errors in Table I that there is a substantial anticorrelation between the kinetic and potential energies, making the statistical errors on the total energy less than either alone. This is typical of a highly quantum system.

The primitive approximation is quite a different matter. One observes that the values obtained converge much slower to the exact results. In fact, we are unable to run the calculations long enough to achieve both a statistical and systematic error of less that 0.25 K.

To estimate the ''best'' results, the values for the limit $\tau \rightarrow 0$, we fit the calculated kinetic, potential and total energies to a power law:

$$E = a + b\tau^c, \tag{16}$$

TABLE I. Energies as computed with DT-PIMC with the pair action for various numbers of time slices M. l is the number of sampling levels. Runs consisted of 500 blocks each of 1000 passes. The equilibration consisted of 100 blocks. Those blocks do not contribute to the averages or the computer time used. The errors were estimated using blocking analysis. K_B stands for the calculation of the kinetic energy with the thermodynamic (Barker) estimator. Energies are in K/molecule, and computer time t in thousands of seconds on a Convex Exemplar S-class system.

M	l	V	δV	K	δK	K_B	δK_B	E	δE	t
8	1	−48.222	0.076	30.524	0.060	30.524	0.068	−17.698	0.040	51
8	2	−48.127	0.065	30.490	0.040	30.535	0.071	−17.636	0.030	60
12	3	−48.625	0.078	30.798	0.052	30.763	0.082	−17.827	0.036	68
16	3	−48.651	0.080	30.923	0.059	30.893	0.075	−17.728	0.038	79
24	3	−48.572	0.092	30.881	0.064	30.847	0.091	−17.690	0.050	97
32	3	−48.573	0.098	30.884	0.066	30.970	0.093	−17.689	0.054	108
40	3	−48.708	0.092	31.030	0.064	30.913	0.101	−17.671	0.050	127

where c was a fixed exponent. The results of the fits are given in Table III. The fits for the pair action used all the data points for kinetic and potential energies, but for the total energy we use only the values with $M \geqslant 12$. In this case, the best fits are given by $c=3$, implying that the error of the energy is proportional to τ^3. This can be understood by examining the ''residual error'' of the action given in Eq. (4.51) of Ref. 2. The fitted values of a are our estimates of the ''exact'' values of the energies. Using b we can estimate the number of time slices necessary to have a systematic error, $\delta \leqslant 0.25$ K. We find that we must have $M \approx 10$ for all three components of the energy. This coincides with the approximate value of 8–12 given by inspecting Fig. 4.

The corresponding value of $\tau \approx 0.0167$ K^{-1}, is smaller than that used for superfluid helium (0.025 K^{-1}).[2] The smaller τ is likely due to the increased particle density in the core of the cluster relative to a He cluster and the fact that the mass of H_2 is half of that of ^{4}He and thus has a larger zero point kinetic energy. The thermodynamic estimator of the kinetic energy gives results, aside from larger errors, consistent with the virial estimate, as it should in a well-converged simulation.

To see if the results of the calculations with the primitive approximation are compatible with those of the pair action approach, we fit the data from Table II with Eq. (16). For all three components of the energy, we neglect the first two data ($M \leqslant 40$) which have a significantly different power from the more accurate values of M. We fixed the a parameter at the previously determined ''best'' value from the pair action, and used $c=2$ (derived from the residual error in the primitive approximation[2]). We find that, in the limit of $\tau \to 0$, the energies converge to the same values computed with the pair action within the larger statistical error bars. Figure 4(a) also shows the potential energies for a larger hydrogen cluster ($N=33$) and lower temperature as computed by Scharf *et al.*[7] (We scaled those results by a factor of 0.72 to account for the different conditions.) This is, to our knowledge, the only complete reference on pure H_2 clusters that reports results at different τ. They used the DT-PIMC method within the primitive approximation for the action. That approximation has also been used to study hydrogen clusters with impurities,[15–17] solid hydrogen[18] and fluid[19] H_2. The errors in the potential energies and its convergence are in good agreement with the present calculations, taking into account the different physical conditions. From our fits, we estimate that the number of slices needed to have a systematic error of the energy less than 0.25 K/molecule in the primitive approximation is $M \sim 190$ for the kinetic, $M \sim 105$ for the potential, and $M \sim 180$ for the total energy.

TABLE II. Energies as computed with DT-PIMC under the primitive approximation. As in Table I, each run consisted of 500 blocks each of 1000 passes (except for the run with $M=200$ which had 250 blocks). The equilibration consisted of 100 blocks. Units as in Table I.

M	l	V	δV	K	δK	E	δE	t
20	3	−55.849	0.100	28.308	0.041	−27.541	0.066	81
40	3	−50.934	0.123	29.328	0.073	−21.607	0.071	124
80	3	−48.756	0.148	29.693	0.098	−19.063	0.070	199
120	3	−48.562	0.148	30.093	0.129	−18.470	0.076	271
200	3	−48.659	0.119	30.539	0.189	−18.119	0.144	166

Shown in Fig. 5 is the statistical efficiency versus τ. Note that the DT-PIMC algorithm allows one to increase the number of time slices (decrease τ) without adversely affecting the efficiency. The two points for $\tau=0.0208$ K^{-1} represent the efficiencies for different number of levels (l): the upper point (more efficient) refers to the calculations with $l=2$, and the lower one represents the data with $l=1$. (The energies from the two levels are the same within the error bars.) This sort of comparison allows one to optimize the algorithmic parameters.

The efficiency of the primitive action is about the same as that of the pair action for equal numbers of time slices as expected. In the case of the kinetic energy, there is even a small gain of efficiency in using the primitive approximation because the action can be computed more quickly. As mentioned above, exactly the same sampling and estimators are used so the comparison is unaffected by other differences in the algorithm. The main difference between these two actions is the slower convergence with respect to the number of time slices, as can be seen in Fig. 4. Thus in Fig. 6, we see the time we need to reach a given combined error with the primitive action is two orders of magnitude longer than with the pair action.

TABLE III. Parameters of the least squares fit of the different data to a form given by Eq. (16). See text for more details. χ^2 is the usual statistic for the fit and ν is the number of degrees of freedom. PA stands for ''primitive approximation.''

	a	δa	b_{DT}	δb_{DT}	c_{DT}	b_{PA}	δb_{PA}	c_{PA}	b_F	δb_F	c_F	χ^2/ν (DT)	χ^2/ν (PA)	χ^2/ν (FPI)
KE	30.93	0.03	-48×10^3	5×10^3	3	-3.3×10^5	2×10^4	2	-101	5	1	0.58	2.1	0.85
PE	-48.68	0.04	55×10^3	8×10^3	3	-1×10^5	7×10^3	2	-3820	180	2	1.04	4.0	1.23
TE	-17.67	0.03	-58×10^3	19×10^3	3	-3×10^5	1.5×10^4	2	-196	9	1	0.03	2.3	1.10

IV. FOURIER PATH INTEGRAL MONTE CARLO

In the Fourier path integral algorithm (F-PIMC), one represents the imaginary time path starting at R and ending at R' at ''time'' β by its Fourier components,[20,21]

$$R(u)=R+(R'-R)(u/\beta)+\sum_{k=1}^{\infty}\mathbf{a}_k\sin(k\pi u/\beta), \tag{17}$$

where $\mathbf{a}_k$ denotes the kth Fourier coefficients. (Each $\mathbf{a}_k$ is a $3N$ vector; one component for each coordinate of the N particles.) An expansion in Fourier components is mathematically equivalent to the discretized path representation;[22] however, the type of approximate actions, sampling and estimators for the energy depend on the representation, enough so that F-PIMC and DT-PIMC can be considered different numerical methods. For calculation, the Fourier series must be truncated at some value m $(k<m)$. This is the discretization parameter corresponding to M in DT-PIMC. In comparing the methods we occasionally set $m=M$; the discretization level is about the same whether we represent the path by M time slices or by m Fourier components. Of course in the comparison of efficiency we find the discretization level needed to reach a given systematic error so that this definition does not affect the results.

One can show[23] that the partition function can be written as an integral over the Fourier coefficients:

$$Z=J\int dR\ d\mathbf{a}\ \exp\left(-\sum_k\frac{\mathbf{a}_k^2}{2\sigma_k^2}-\beta V_{\mathrm{eff}}\right), \tag{18}$$

where J is a constant, and V_{eff} is the path-averaged potential energy:

$$V_{\mathrm{eff}}=\int_0^{\beta}\frac{du}{\beta}\ V[R(u)]\approx\frac{1}{Q}\sum_{j=1}^{Q}V[R(j\beta/Q)]. \tag{19}$$

This potential is evaluated by a Q-point trapezoidal quadrature; Q is an additional algorithmic parameter in the simulation. The Fourier components have a mean-squared fluctuation given by

$$\sigma_k^2=\frac{4\beta\lambda}{\pi k^2}. \tag{20}$$

The integrand of Eq. (18) is used for the action of a path. The variables to be sampled with a Metropolis procedure are the $3N$ spatial coordinates R and the $3Nm$ Fourier coefficients $\mathbf{a}_k$.

A. Partial averaging

In our simulations we have used the computationally most efficient F-PIMC algorithm currently in use which incorporates partial averaging.[21] In this approach, one approximately accounts for the neglected Fourier modes, those with $k>m$, by correcting the action. These short wavelength modes are relatively decoupled from the potential and, to a good approximation, are pure Gaussian fluctuations about the path that is determined by the lower-order Fourier coefficients that are explicitly carried along in the simulation. The width of the short wavelength fluctuations is

$$\sigma^2(u)=\frac{2\lambda u}{\beta}(\beta-u)-\sum_{k=1}^{m}\sigma_k^2\sin^2\left(\frac{k\pi u}{\beta}\right). \tag{21}$$

Expanding the potential to second order in displacements with respect to the path generated by the Fourier components with $k\leqslant m$ the action is corrected by defining the effective potential as

$$V_{\mathrm{eff}}[R(u)]=(1/\beta)\int_0^{\beta}du\left(V[R(u)]+\sum_{i=1}^{3N}0.5\sigma^2(u)V_{ii}[R(u)]\right), \tag{22}$$

where $V_{ii}[R(u)]$, $i=1$ to $3N$, is the diagonal element of the second-derivative matrix, or the Hessian, evaluated at $R(u)$. One replaces the path average of the bare potential by the path average of a quantum corrected potential. We tested what happened when we used a effective mass [doubling λ in Eqs. (20) and (21)] rather than the bare mass with the number of Fourier components $m=16$, but the results were not systematically improved.

B. Estimators

We use two different estimators for the potential energy, $\langle V\rangle$, and $\langle V_{\mathrm{eff}}\rangle$. They must be equal in the limit $m\to\infty$. We use the thermodynamic (T) estimator for the total energy, obtained by analytically differentiating Eq. (18) with respect to β. The kinetic energy is obtained by subtracting the potential energy from the total energy. Two other estimators for the kinetic energy have been suggested in the context of F-PIMC simulations. The virial estimator was found to have relatively large errors when tested in F-PIMC simulations.[20] The direct (H) estimator, in which one evaluates the expectation of the kinetic operator on the density matrix, requires evaluation of derivatives of the potential and has not been used in conjunction with partial averaging for clusters or liquids. It is possible that the use of the direct estimator could dramatically decrease the variance of the kinetic energy[20] but because this estimator has not been used on a problem of this sort, we have chosen to test only the thermodynamic estimator.

TABLE IV. Kinetic, potential and total energies computed with the F-PIMC method as a function of the number of Fourier components m and quadrature points Q. The number of passes is 1.4×10^5 for $m\leqslant 8$, 2.8×10^5 for $m=16$, and 4.65×10^5 for $m\geqslant 24$; a pass being defined as a single attempted move of all the molecules. Units as in Table I.

m	Q	V	δV	K	δK	E	δE	t
4	12	−55.31	0.29	26.86	0.22	−28.45	0.17	2
8	24	−50.43	0.36	28.54	0.29	−21.87	0.23	4
16	48	−49.69	0.19	29.92	0.29	−19.76	0.22	12
24	72	−48.72	0.23	29.87	0.26	−18.85	0.25	27
30	90	−48.48	0.20	30.32	0.39	−18.16	0.33	33
36	108	−48.54	0.19	30.16	0.35	−18.39	0.34	39

C. Sampling the paths

Metropolis Monte Carlo was used to perform the integrals as is done in the DT-PIMC method. However, the algorithm to move the paths was somewhat different. The basic step consists of an attempt to change all the Fourier components of a single particle and its center of mass $\mathbf{r_i}$ simultaneously. The new configuration is accepted or rejected based on the change in action.

Both the center of mass and the Fourier components are moved uniformly in a cube about the old coordinates:

$$\mathbf{r_i'}=\mathbf{r_i}+\Delta_{\rm cm}(\mathbf{u}-0.5) \tag{23}$$

and

$$a_k'=a_k+\Delta_f\sigma_k(\mathbf{u}_k-0.5), \tag{24}$$

where the $\mathbf{u}$ are independent uniformly distributed random numbers in (0,1). Typical values for the maximum displacement were: $\Delta_{\rm cm}=0.26$ Å and $\Delta_f=0.15$ Å for $m=36$.

To summarize, the algorithmic parameters in F-PIMC are the number of Fourier coefficients m which controls the extent of fluctuations in the quantum paths, the number of points Q used to integrate the potential, and the parameters $\Delta_{\rm cm}$ and Δ_f which control the movements through path space. By studying the convergence of the MC averages for a given value of m as a function of Q, we found the value of Q necessary for convergence to within a given systematic error. For quasi-classical systems like neon, we have found $Q=m$ to be sufficient but for hydrogen we found that $Q=3m$ was necessary (for convergence to within 1.5%). Then $\Delta_{\rm cm}$ and Δ_f were set to achieve an acceptance ratio of roughly 50%. For the purposes of comparing with the DT-PIMC a *pass* is defined as $N=22$ such attempted moves. We tested our algorithm by comparing the results for a system of seven neon atoms against the energies, given as a function of the number of Fourier coefficients, in Ref. (24) and obtained agreement.

D. Results

Table IV shows the energies and error bars versus the number of Fourier components for the hydrogen cluster. As before, we fit the results to a power law in $\tau_F=\beta/m$; see Eq. (16). Since the statistical errors from the F-PIMC runs were larger than those from the DT-PIMC we assumed that the ''best'' values for the energies were given by the a's previously determined and asked whether a ''reasonable'' fit could be obtained by varying only b and c. We used all points for the fit, except in the case of the total energy, in which we drop the value with $m=4$. The optimal parameters and χ^2 for the fits are given in Table III. We find that the kinetic and total energies converge linearly in τ_F while the potential energy converges quadratically. We know of no analytic derivation for how the Fourier path partial averaging procedure should converge. However, as one can see by the values of χ^2/ν, the fits to the energies are reasonable, confirming that the different methods are converging to the same results. Unpublished results of Doll on more classical clusters suggest an exponent of 2 ($c=2$), but that is not consistent with the present calculations. Several factors could explain the inconsistency, including the different values of m used for fitting and differences in the quadrature evaluation.

As seen in Fig. 4, the errors converge much slower with the F-PIMC method than with the DT-PIMC (using the pair action) but better than the DT-PIMC using the primitive action. Using the fit parameters given in Table III, the value of m needed to meet the requirements of systematic error $\delta\leqslant 0.25$ is found to vary from 21 for the potential energy to 131 for the total energy. To reduce the systematic error on the kinetic energy to the needed range, we need $m=67$. The number of Fourier components is much larger than the number needed for the pair action in the DT-PIMC method but much less than the primitive approximation requires. The efficiency and convergence (time needed to obtain a certain total error) of the F-PIMC method for the various components of the energy are shown in Figs. 5 and 6.

V. COMPARISON AND SUMMARY

We have found that the DT method with the pair action converges faster with τ (τ^3 vs. $\tau_F^{1,2}$) and it is more accurate (the statistical errors are smaller) than the F-PIMC. This is made precise in Figs. 5 and 6. For example, Fig. 5 shows that for a given discretization parameter the DT-PIMC has a higher efficiency. (Note the log–log scales.) The ratio is fairly modest in the case of the potential energy (say, a factor of 2), but much larger for the kinetic energy. For the kinetic energy, the slopes are different: the decrease of the efficiency with the increase in the number of Fourier components is faster than the decrease when the number of time slices grows. This is likely due to the use of the virial estimator for the kinetic energy.

However, an even more significant effect is given by rate of convergence of the ''true error:'' the time needed to get both systematic and statistical errors less than a given amount versus that error. The simplest quantities to evaluate in both types of path integral simulations are observables involving coordinates only such as the potential energy. The convergence is rapid in both methods: the potential energy converges for ten time slices (Pair action DT-PIMC) or 21 Fourier coefficients (F-PIMC). The DT-PIMC (with pair action) for a target error of 0.25 K is about 3 times faster for the potential energy. The convergence behavior of the potential energy is expected to be representative of the behavior of

the expectation value of operators which are diagonal in the coordinates.

However, the kinetic energy converges much slower. About 300 times as much computer time is needed to reach the target error in the F-PIMC method. The kinetic energy depends on adequately accounting for the small fluctuations in the paths in steeply varying regions on the potential, e.g., when two particles closely approach. This is typical for any strongly nonlinear potential. The DT-PIMC method with the pair action appears to be much more efficient in evaluating the kinetic energy: ten time slices are required instead of 67 Fourier coefficients. Adding these components increases the computer time drastically. Partial averaging initially improves the convergence; about 24 Fourier coefficients are require to get within 97% of the asymptotic value, but it appears to be slow in picking up the last few percent in the kinetic energy. As discussed later, this may be a special difficulty associated with systems with very steeply varying repulsive pair potentials.

The computer time requirements for the DT-PIMC with the primitive action are significantly worse than for the F-PIMC method. By comparison with the results of the primitive approximation, we can see that most of the efficiency of the DT-PIMC method is due to the better treatment of the action, not to the actual representation in time slices or to the sampling method.

We cannot be sure that some of the difference in computer time is not due to factors in the particular implementation of the different methods in the corresponding computer codes. Neither code was specialized for this problem or computer. With another implementation of the algorithms, the ratio of efficiencies could be either greater or less than the 2 we observed for the efficiency of evaluating the potential energy. However, no amount of programming skill can account for the factor of 300 in the time to compute the kinetic energy to a given total error.

The choice of the kinetic energy estimator is one important deficiency in the F-PIMC method which our study has revealed. In the present F-PIMC calculation, the thermodynamic estimator of the total energy has been used which, though simple to implement, has a very serious drawback in that its variance grows proportionally to the number of Fourier components.[20] The variance of this estimator also grows with the number of time slices in DT-PIMC, as can be seen in the errors of the kinetic energy, K_B, given in Table I. Even though both are increasing overall, the variance in DT-PIMC is considerably lower than in F-PIMC. The direct (H) estimator in some applications has lower variance and possibly different systematic errors, but has not been implemented together with partial averaging. An important problem is to accelerate the convergence of the kinetic energy to the exact value. Convergence to within 3% of the asymptotic value is achieved with 24 Fourier coefficients, which corresponds to a computational efficiency within a factor of 3 of the DT-PIMC algorithm. Forty more coefficients have to be added to obtain the last few percent and result in slowing down the program by a factor of 100. This suggests that it may be worthwhile to attempt to improve the partial averaging algorithm; the quantum corrected potential includes only the second-order term and it may be necessary to go beyond this. However, such expansions for systems interacting with strong potentials typically only converge asymptotically, and a more detailed study is necessary to establish whether higher-order corrections will improve computational efficiency. The sampling strategy used in F-PIMC is very simple. Improvements are clearly necessary in order to reduce correlations in the MC walk and to enhance its applicability at lower temperatures. However Fig. 5(b) demonstrates that the sampling scheme employed is adequate for this problem; less than a factor of 2 improvement could be gained by a more efficient sampling. However, for a more quantum mechanical system, significant tuning of the F-PIMC sampling algorithm would be needed. In this regard there is one fundamental difference in the two algorithms. In DT-PIMC, it is possible to make moves local in imaginary time, while in F-PIMC movement of the Fourier path coefficients necessarily means that the entire time integral needs to be redone. If the key output desired from a simulation involves mainly coordinate-dependent observables, then the simplicity and adaptability of the F-PIMC algorithm is an attractive feature.

In DT-PIMC, convergence can be improved by including polarization terms, in particular, the off-diagonal polarization. That would be significantly slower per step, so it is not clear that adding many-body terms in the action will be compensated for by fewer time slices. Performance can also be improved by optimizing to use the architecture of the current generation of computers. In addition, sampling of phase space can probably be done better by using correlation or special tricks for a cluster such as preferentially sampling atoms on the surface.

Finally, we comment on the extent can we generalize from this comparison to the simulation of other quantum problems. First, consider the case of a more classical or higher temperature system. There errors in the total energy will be dominated by the fluctuations in the potential energy and not by quantum effects. The errors can only be reduced by moving through phase space quicker. The virial estimator explicitly computes the quantum correction to the classical kinetic energy directly with low variance. However, most path integral calculations of nearly classical systems will not be so time consuming that advanced quantum simulations methods will be needed.

In lower temperature quantum systems, exchange effects become physically important, leading to such phenomena as superfluidity, formation of a Fermi surface, and so forth. This necessarily involves sampling paths where the particles permute labels upon path closure, making the sampling methods more complicated. These PIMC sampling methods are successful in being able to simulate bosonic superfluids and Fermi liquids as long as the temperature is not too low.[25] There is very little experience on systems where quantum exchange is important using F-PIMC.[23]

One of the open questions of our comparison is how dependent are the results on the assumed interaction potential. For problems with particles interacting strongly with pair potentials, the pair density matrix is quite accurate. If we had done the comparison with a smooth potential, say a har-

monic oscillator, or a many-body potential not equal to a pairwise sum, e.g., the potential of many water molecules taking into account the hydrogen and oxygen atoms, perhaps a representation in Fourier paths with partial averaging would be more accurate. The action for those potentials may be hard to represent in coordinate space. Such cases remain to be investigated.

We hope that this paper, even with the limitations in the comparison described, stimulates a reexamination of methods for path integral simulations. It is an extremely powerful method, but, as we have shown, it can be an extremely slow, particularly as accuracy requirements increase. There have been recent developments to calculate electronic forces (using density functional theory) in a path integral molecular dynamics simulation of the protons for small protonated clusters.[16,17] These calculations have been done with the primitive approximation. Given the computer time requirements, it is clearly important to use the most effective path integral technique available. A program to compute the pair action is on the web site *http://www.ncsa.uiuc.edu/Apps/CMP/index.html* . This is the essential ingredient needed for more accurate results for atomic systems which have characteristic strong, short-range repulsive interactions.

ACKNOWLEDGMENTS

This work was supported by NSF DMR94-224-96, the Department of Physics and NCSA at the University of Illinois, the Spanish Ministry of Education and Culture, and the Department of Science and Technology, India (Grant No. SP/S1/H-36/94).

[1] R. P. Feynman and A.R. Hibbs, *Quantum Mechanics and Path Integrals* (McGraw–Hill, New York, 1965).
[2] D. M. Ceperley, Rev. Mod. Phys. **67**, 279 (1995).
[3] M. H. Kalos and P. A. Whitlock, *Monte Carlo Methods, Volume I: Basics* (Wiley, New York, 1986).
[4] M. P. Allen and D. J. Tildesley, *Computer Simulations of Liquids* (Oxford University Press, Oxford, 1987).
[5] P. Sindzingre, D. M. Ceperley, and M. L. Klein, Phys. Rev. Lett. **67**, 1871 (1991).
[6] C. Chakravarty, Mol. Phys. **84**, 845 (1995).
[7] D. Scharf, G. J. Martyna, and M. L. Klein, J. Chem. Phys. **97**, 3590 (1992).
[8] I. F. Silvera and V. V. Goldman, J. Chem. Phys. **69**, 4209 (1978).
[9] C. Chakravarty, J. Chem. Phys. **102**, 956 (1995).
[10] M. Wagner and D. M. Ceperley, J. Low Temp. Phys. **102**, 275 (1996).
[11] J. A. Barker, J. Chem. Phys. **70**, 2914 (1979).
[12] R. G. Storer, J. Math. Phys. **9**, 964 (1968).
[13] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, J. Chem. Phys. **21**, 1087 (1953).
[14] B. J. Berne and D. Thirumalai, Annu. Rev. Phys. Chem. **37**, 401 (1986).
[15] D. Scharf, G. J. Martyna, and M. L. Klein, J. Chem. Phys. **99**, 8997 (1993).
[16] I. Stich, D. Marx, M. Parinello, and K. Terakura, Phys. Rev. Lett. **78**, 3669 (1997).
[17] I. Stich, D. Marx, M. Parinello, and K. Terakura, J. Chem. Phys. **107**, 9482 (1997).
[18] M. Zoppi and M. Neumann, Phys. Rev. B **43**, 10 242 (1991).
[19] D. Scharf, G. J. Martyna, and M. L. Klein, Fiz. Nizk. Temp. **19**, 516 (1993).
[20] J. D. Doll, D. L. Freeman, and T. L. Beck, Adv. Chem. Phys. **78**, 61 (1990).
[21] R. D. Coalson, D. L. Freeman, and J. D. Doll, J. Chem. Phys. **85**, 4567 (1986).
[22] R. D. Coalson, J. Chem. Phys. **85**, 926 (1986).
[23] C. Chakravarty, J. Chem. Phys. **99**, 8038 (1993).
[24] S. W. Rick, D. L. Leitner, and J. D. Doll, J. Chem. Phys. **95**, 6658 (1991).
[25] D. Ceperley, Phys. Rev. Lett. **69**, 331 (1992).