Journal of Statistical Physics, Vol. 72, Nos. 3/4, 1993

Molecular Dynamics and Time Reversibility

D. Levesque[1] **and L. Verlet**[1]

Received December 17, 1992; final April 1, 1993

We present a time-symmetrical integer arithmetic algorithm for numerical (molecular dynamics) simulations of classical fluids. This algorithm is used to illustrate, through concrete examples, that time-asymmetric evolutions are typical for systems of many particles evolving according to reversible microscopic dynamics and to calculate the asymptotic behavior of the velocity autocorrelation function with an improved accuracy. The equivalence between equilibrium time averages and microcanonical ensemble averages is checked via two new sampling methods for computing microcanonical averages of classical systems.

KEY WORDS: Numerical simulations; irreversibility; Monte Carlo methods; microcanonical ensemble; long-time tail.

1. INTRODUCTION

Ludwig Boltzmann gave a clear and convincing explanation of how macroscopic irreversibility arises from the reversible Newtonian equations operating at the microscopic level. His well-known solution to that problem is based on the very large number of particles ($N \gtrsim 10^{20}$) involved in the evolution of macroscopic systems. Boltzmann argued that, for such large N, any initial state corresponding to a nonequilibrium macroscopic situation will move toward states of increasing Boltzmann entropy, i.e., toward equilibrium (for a recent, comprehensive and lucid review see Lebowitz(1)). While Boltzmann's point of view is now accepted by most physicists, it does have some prominent vocal opponents (e.g., Karl Popper(2) and Ilya Prigogine(3)) and, as pointed out in ref. 1, there is still an alarming amount of confusion about "the problem of irreversibility." In particular there are questions of whether the observed time asymmetry in "real life" may not be

---

[1] Laboratoire de Physique Théorique et Hautes Energies (Laboratoire associé au Centre National de la Recherche Scientifique), Université de Paris XI, 91405 Orsay Cedex, France.

0022-4715/93/0800-0519$07.00/0

due to some inherent microscopic time-asymmetric laws already discovered (such as those present in the weak interaction) or yet to be discovered. Another objection is that physical systems are never truly isolated and moreover that no one really understands fully how such nonisolated systems behave in a quantum mechanical world.

It might have been thought that computer simulations should provide a privileged way for illustrating Boltzmann's point of view. These simulations are made on deterministic and finite computers. They rely on algorithms which, in general, are explicitly time-reversible. They clearly show approach to equilibrium for large systems. The fact that in simulations N is not as large as in real macroscopic systems is not really a big problem. For the values of N considered in numerical simulations (a few hundred or few thousands) the recurrence time (Poincaré time) of any particular microscopic state is overwhelmingly larger than the time of the universe, so that such a recurrence is practically unobservable in any computer calculation.

It turns out, however, that, due to rounding-off errors, the actual simulations performed up to now are not strictly time-reversible and therefore cannot help to dissipate the above-mentioned confusion. In a classic paper,(4) Orban and Bellemans considered the evolution of a system of colliding hard disks. Although the equations of motion for such a system are fundamentally reversible, the computed trajectories obtained when, at a given time, all velocities are reversed do not coincide with those traced by the direct motion. The reason for this discrepancy is that, when the floating-point arithmetic of any computer is used for a simulation (as is generally done), the rounding-off errors may not be the same in the direct and time-reversed computations. Due to the instability of the dynamical equations with respect to small perturbations, those tiny differences propagate and increase exponentially. Orban and Bellemans provided a dramatic illustration of this effect by computing a Boltzmann H-function for a hard-disk system starting from an ordered configuration and comparing it to the same function after a time-reversal: the H-function corresponding to the reversed motion did rise when the origin of time was approached from above, but less than it should have, because the memory of the original correlations was partly lost.

The first purpose of the present paper is to show that the computer simulations, which have proven their relevance in predicting the actual behavior of simple and complex many-particle systems, can be also used to support Boltzmann's interpretation when they are properly carried out. The integration algorithm proposed by one of us(5) and generally used in molecular dynamics (MD) computations is revisited. After a brief historical digression, we proceed to improve the computer implementation of this algorithm, which is explicitly time-symmetrical, but is subject, in specific computations, to rounding-off problems similar to those met by Orban and Bellemans. We show that, by a suitable use of integer arithmetic operations, those unwanted numerical effects are entirely suppressed. The modified implementation of the algorithm is about as efficient as the unmodified one. It has the further advantage of yielding an exact conservation of total momentum and of offering the theoretical guarantee that no systematic drift of the total energy should occur.

Several long computations have been performed using this new algorithm so as to test that the reversed motion trajectories exactly coincide with the direct ones, both by calculating the corresponding H-functions, which turn out to be identical, and by checking directly that the endpoints of the reversed trajectories exactly coincide with the corresponding origins of the direct trajectories. These long simulations have also allowed us to verify that the present changes of the MD algorithm do not induce a small but monotonic increase or decrease of the total energy.

The computations are all performed starting from an unstable ordered initial configuration, which immediately melts. As remarked by Lebowitz in the above-quoted paper, the equilibrium state which is reached thereafter and remains in the rest of the direct computation is quite peculiar. When integrated forward, it looks quite normal, and in particular it leads to an asymptotic time decay $1/t^{3/2}$ for the velocity autocorrelation function. This function is calculated here for a continuous potential with a precision which, improved by a factor of 3 as compared with the previously known results,(6) is now similar to that obtained for a system of hard spheres.(7) But when, at any time, velocities are reversed, those "normal" equilibrium trajectories are traced back to the highly ordered initial state which has been exactly memorized. If the time goes on in the reverse direction, the "solid" melts again and different equilibrium trajectories are obtained, which contain also the exact memory of the ordered state which they met on their path.

Another property of equilibrium trajectories is that, in the phase space, they cover up densely and uniformly a surface of states of identical energy, so that time averages of microscopic quantities are equivalent to microcanonical ensemble averages. In order to illustrate properly this equivalence, we must be able to compute those ensemble averages. Such computations are performed by two different Monte Carlo (MC) methods proposed by Ray(8) and by Creutz.(9) We show that the method devised by Ray can be cast into a form which makes it very similar to the standard Metropolis(10) method (the only difference is that, instead of being fixed, the temperature is calculated, at each step of the sampling process, from the kinetic energy defined as the difference between total and potential energies). The Creutz algorithm derived for a system of spins on a lattice is extended to the case of classical fluids. It is modified so as to keep both the total energy and momentum fixed.

2. TIME-REVERSIBLE ALGORITHM

In order to integrate Newton's equation of motion of the ith molecule located at position $\mathbf{r}_i$ at time t, we use the algorithm

$$\mathbf{r}_i(t+h) = -\mathbf{r}_i(t-h) + 2\mathbf{r}_i(t) + h^2\mathbf{F}_i(t) \tag{1}$$

where $\mathbf{F}_i$ is the sum of the forces on that molecule divided by its mass and h is the time step of integration. This time-symmetrical algorithm$^{(5)}$ is widely used in molecular dynamics computations, where the steeply varying forces create a specific problem.

The algorithm (1) was proposed by one of us as a part of a systematic effort aimed at reducing the computer time of the MD simulations. In his admirable pioneering work, Rahman$^{(11)}$ used, for integrating Newton's equations, the Runge-Kutta method, a predictor-corrector method which breaks time symmetry. In order to build a time-symmetrical integration scheme, we simply added the Taylor expansions of $\mathbf{r}_i(t+h)$ and of $\mathbf{r}_i(t-h)$ so as to get an expansion scheme in even powers of h. We then thought for a short while that the complicated three-body terms involved in the evaluation of the h^4 term of the scheme could be approximated, but, as should have been evident to us from the start, that term must be computed with utmost care in the case of steeply varying potentials such as the Lennard-Jones potential. When we finally decided to keep algorithm (1) as such, we realized that it was simply the finite-difference transcription of Newton's equation and we did not bother to search for its first occurrence in the literature, because of its triviality. We recently discovered that it had been explicitly written and used by the astronomer Joseph Delambre in 1791. Unfortunately, Delambre clearly considered this algorithm as too obvious to give any reference to previous literature.$^{(12)}$

It should be added that some standard textbooks on numerical analysis (e.g., ref. 13) give a powerful scheme devised at the beginning of the present century so as to integrate accurately Newton's equation in the case of smooth interactions.$^{(14)}$ This scheme, due to Störmer,$^{(15)}$ has been shown to be outstandingly good in that case and has been successfully applied to astronomical and electromagnetic computations. In Störmer's scheme, the correction terms in the Taylor expansion considered above are calculated through finite differences: this entails a break in the time symmetry which has no consequence in the cases for which the method was devised, but would have catastrophic effects in the case of intermolecular interactions. It is thus misleading as well as unfounded from a historical standpoint to associate, as is sometimes done,$^{(16)}$ the simple algorithm (1) used in the MD simulations with the sophisticated scheme invented by Störmer.

The reason why the obvious time symmetry of algorithm (1) is broken in actual computations where the positions and forces involved in (1) are taken as floating-point numbers is rather trivial. It stems from the fact that, according to the direction of the time arrow, the rounding-off procedure currently used in computers (which is practically unpredictable and acts as a very small random noise) may lead to different results whenever, during the integration process described by formula (1), the previous coordinate, that is, either $\mathbf{r}_i(t-h)$ or $\mathbf{r}_i(t+h)$, according to the direction of the time arrow, is subtracted from quantities depending only on t. This eventual break in the time symmetry could be removed by biasing systematically the rounding-off procedure.

A simpler and better method consists in using integer arithmetic for the crucial part of the computation: if $\mathbf{r}_i(t+h)$, $\mathbf{r}_i(t)$, $\mathbf{r}_i(t-h)$, and $h^2\mathbf{F}_i(t)$ are represented by integers, the finite-difference algorithm (1) will stay exactly time-reversible. With this aim in view, we choose the side of the cubic box (with periodic boundary conditions) in which the particles are enclosed as our unit of length. With this unit of length, we associate a large integer equal to 2^{60} (about 10^{18}). This guarantees that the distances between molecules are calculated with a very high degree of accuracy (one part in 10^{17}–10^{18}).

The use of an integer representation of the distances does not need to be extended to the computation of the forces, which depends only on time t. In fact, it is convenient to convert the interatomic distances at time t in a floating-point representation in order to compute easily the partial force $\mathbf{f}_{ij}(t)$. Then $h^2\mathbf{f}_{ij}(t)$ is converted in the integer representation and is used to compute $h^2\mathbf{F}_i(t)=\sum_j h^2\mathbf{f}_{ij}(t)$. As we have said in the introduction, the use of integer arithmetic for expressing the partial forces has the further advantage that it remedies a defect inherent to all MD algorithms performed in floating-point arithmetic. Because adding the same numbers in a different order may give rise to different rounding-off errors, there is no guarantee that the forces $\mathbf{F}_i(t)$ are calculated in such a way that their sum is zero and, consequently, that the total momentum is conserved.

As MD practitioners know, there is, in the course of an actual computation, a slight drift in the total momentum, which in turn entails a drift in the total kinetic energy of the system. In conventional MD computations, this effect may be minimized by subtracting out the spurious momentum introduced by the rounding-off procedure, but it cannot be consistently suppressed. The problem entirely disappears with our method because, since $\sum_i h^2\mathbf{F}_i$ is exactly zero, the total momentum is exactly conserved. The only remaining weak point of the method is that the total energy is not exactly conserved. But, as was mentioned in the introduction, the exact time reversibility of the integration algorithm precludes any systematic drift in the total energy, because the same drift should also appear in the time-reversed motion, which leads to an evident contradiction (this remark is due to Jean-Michel Caillol). A Fortran program, given in Appendix A, shows how the present MD algorithm was implemented.

In order to check our MD algorithm, we performed several computations for systems of particles interacting through the repulsive part of the Lennard-Jones potential:

$$\begin{aligned} v(r) &= 4\varepsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right] + \varepsilon, \qquad && r < 2^{1/6}\sigma \\ &= 0, && r < 2^{1/6}\sigma \end{aligned} \tag{2}$$

Three computations were made, for systems of 108, 256, and 4000 atoms, where the time reversal was done at time $t_r = 2 \times 10^5 h$, $10^5 h$, and $10^4 h$, respectively. The integration step is taken to be $0.0046188(\sigma^2 m/\varepsilon)^{1/2}$, where m is the mass of the atoms. When the evolution time reaches its maximum value t_r, the vectors $\mathbf{r}_i(t_r + h)$ and $\mathbf{r}_i(t_r - h)$ are interchanged and the integration process is carried out the atoms moving backward, until the initial configuration is recovered, at time $t = 2t_r + h$. This initial configuration is that of an fcc perfect crystal with a density corresponding to $\rho\sigma^3 = 0.45$. To complete the initial conditions, we define the configuration at time $t = -h$ by adding to the positions at time $t = 0$ small random displacements drawn from a Gaussian probability distribution, the variance of this distribution being proportional to the initial kinetic energy of the system. As usual, the velocities are calculated from the formula $\mathbf{v}_i(t) = [\mathbf{r}_i(t+h) - \mathbf{r}_i(t-h)]/2h$, where floating-point arithmetic is used. The time reversibility of the algorithm is readily checked by computing the quantity I, where the components of $\mathbf{r}_i$, x_i, y_i, and z_i are integer numbers:

$$I = \sum_i |x_i(0) - x_i(2t_r + h)| + |y_i(0) - y_i(2t_r + h)| + |z_i(0) - z_i(2t_r + h)| \tag{3}$$

which should exactly vanish.

Exploiting a mistake made in a preliminary version of the program, we could show that, when floating-point arithmetic is used, the possible rounding-off error induced by substituting, at the instant t_r when the time is reversed, $\mathbf{r}_i(t_r)$ or $\mathbf{r}_i(t_r + h)$ for one of their 26 corresponding values in the neighboring replicas of the simulation cell is a sufficient cause to drive the system toward an entirely different trajectory.

In Table I, we give the thermodynamic properties of the three systems considered here (108, 256, and 4000 atoms). We also give the maximum difference between the total energy in the initial state and that obtained in the whole course of the computation. For the sake of comparison with Orban and Bellemans, we have computed, at each integration step, the H-function of Boltzmann normalized to 1 at time $t=0$:

$$H(t)=\frac{\int f_t(v)\log f_t(v)\,d\mathbf{v}}{\int f_0(v)\log f_0(v)\,d\mathbf{v}} \tag{4}$$

where $f_t(v)$ is an unnormalized probability distribution of the modulus v of the atomic velocities at time t.

Due to the finite number of atoms, $f_t(v)$ fluctuates around the Maxwell distribution when the system is in an equilibrium state. The time evolution of $H(t)$ is plotted in Fig. 1. The reversible character of the trajectories is clearly shown by the behavior of $H(t)$ for times in the neighborhood of $2t_r+h$. At that time, the value 1 is exactly recovered, in contrast with the computation of Orban and Bellemans. The function $H(t)$ keeps constant and equal to 1 for a few steps. The explanation of this occurence is straightforward: the initial configuration which is recovered by

Table I. Thermodynamic Properties of the Systems Considered[a]

	N	T^*	U	C_V	E_i	E_f	E_m	E_x	E_n	t_r^*	t^*
MD	108	0.975	0.213	1.73	1.6777	1.6772	1.6769	1.6790	1.6748	2	8
MD	256	1.701	0.380	1.72	2.9316	2.9311	2.9314	2.9339	2.9299	1	4
MD	4000	1.617	0.358	1.71	2.7854	2.7852	2.7850	2.7854	2.7848	0.1	0.4
MD	4000	2.176	0.481	1.72	3.7434	3.7453	3.7447	3.7459	3.7434	—	4
MC	4000	2.179	0.481	1.72	—	—	3.7487	—	—	—	0.2
MC	4000	2.183	0.470	1.69	—	—	3.7457	—	—	—	0.96

[a] Columns 1–5, respectively, give for the simulations discussed in the article the type MD or MC, the number N of atoms used in the run, the temperature $T^*=k_B T/\varepsilon$, the potential energy U, and the specific heat C_V. For the MD simulations, E_i and E_f are the total energies of the first and last steps of the run, E_m is the average of the total energy in the run, and E_x and E_n are the maximal and minimal total energies during the run. t_r^* indicates the time step (divided by 10^5) where the time reversal was done. t^* is the total number (divided by 10^5) of time steps in the MD runs. These long MD runs show that there is no systematic increase or decrease of the total energy in the present MD algorithm. For the MC runs (sixth line: Ray procedure; last line: Creutz procedure), the last column gives the number of trial moves per atom (divided by 10^5). The statistical errors are smaller than 1% for the MD runs, about 1% for the MC(Ray) run and $\sim 2\%$ for the MC(Creutz) run. For all runs, the density is $\rho\sigma^3=0.45$.

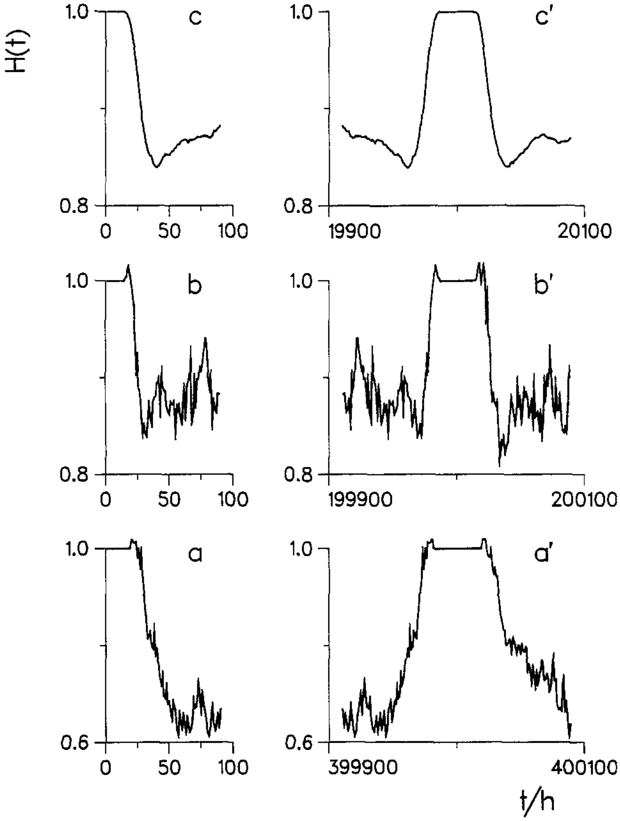

Fig. 1. Parts (a) and (a′) illustrate the time evolution of the Boltzmann H-function $H(t)$ [Eq. (4)] obtained by a time-reversible simulation of a 108-atom system where time reversal is done at the step 200,000. Part (a) gives the plot of $H(t)$ for the first 90 time steps and part (a′) for the 90 time steps before and after the time step 400,000 where the system has gone back to the initial configuration. Parts (b, b′) and (c, c′) give similar plots for systems of 256 and 4000 atoms. In these simulations the time reversal is done at the time steps 100,000 and 10,000, respectively and the systems have gone back to their initial configurations at the steps 200,000 and 20,000. The reversibility of the trajectories is manifested by the fact that the left parts of the plots (a′, b′, c′) are mirror images of the graphs (a, b, c). The results for the 256-atom and 108-atom systems illustrate the fact that the fluctuations are larger in small systems. The initial configuration of the 108-atom systems was chosen more "ordered" than for the two other systems: the initial positions were on an fcc lattice and all the moduli of the velocities were identical.

the time reversal corresponds to an artificial low-density crystal for which, due to the shortness of the range of the potential, the atoms do not interact with each other. $H(t)$ looks symmetrical around $2t_r + h$, but this symmetry is only approximate, as it should be. The computation was carried out up to time $t = 4t_r$: obviously the system went back to its equilibrium state.

Using the above-described MD algorithm, we have preformed an accurate computation of the asymptotic behavior of the velocity autocorrelation function (VAF) without having to worry about small fluctuations in the total momentum. We have carried through 20 runs of 2×10^4 integration steps [with a step size $h = 0.0046188(\sigma^2 m/\varepsilon)^{1/2}$] for a system of 4000 atoms at $\rho\sigma^3 = 0.45$ and $k_B T/\varepsilon = 2.176$. The potential energy is then equal to 0.4810ε and the specific heat at constant volume C_V to $1.72k_B$. Averaging over these 4×10^5 integration steps, we find that the average total energy per atom is 3.7447ε, the initial energy being equal to 3.7434ε, the final value to 3.7453ε, the maximum value reached during the computation to 3.7459ε, and the minimum value to 3.7434ε. As may be seen from Table I, where similar data for other computations are given, the apparent rise of the final total energy above the initial one is not typical.

The computed VAF is given in Table II with the estimated standard error. The present results should be compared with those obtained by Levesque and Ashurst.(6) They are in good agreement with theirs,[2] but somewhat more precise (the standard error on the VAF is reduced by a factor 3). As Table II shows, the asymptotic value of the VAF (which is of the form $\alpha/t^{3/2}$) is reached for $t > 640h$, a higher value than was estimated from the available data in ref. 6, where the value $t > 460h$ was given. This largely reduces the range available to estimate the asymptotic decay of the VAF, as, due to sound waves, times above $800h$ cannot be used. Within one standard error, the product $f(t) \times t^{3/2}$ is constant for $t > 640h$. The value of $\alpha = 0.018$ is about 10% smaller than the value estimated in ref. 6 (using a time unit differing by a factor $48^{1/2}$). This value is in better agreement with the theoretical value 0.017 which can be estimated from the diffusion coefficient and the kinematic viscosity through the formula $\alpha = 2k_B T/m[4\pi(D + \nu)]^{3/2}$, where D is the self-diffusion coefficient and ν the kinematic viscosity. As a concluding remark, it should be noticed that the present time-reversible MD algorithm can advantageously be used on all computers where the integer arithmetic is vectorized whenever the exact conservation of momentum and the improvement in energy conservation are sought. The price to pay is a slight increase (about 10–20%) of the CPU elapsed time.

---

[2] A misprint should be corrected in that paper: the value of the potential energy should read 0.4812. It should also be noticed that C_V was not computed directly as it is here, but was obtained using perturbation theory.

Table II. Velocity Autocorrelation Function and Standard Error[a]

t	$f(t)$	σ_s	$f(t)\,t^{3/2}$	σ_s
0.0000	0.1000 + 01	0.00 + 00	0.0000 + 00	0.00 + 00
0.0462	0.8827 + 00	0.39 − 04	0.8762 − 02	0.38 − 06
0.0924	0.6910 + 00	0.93 − 04	0.1940 − 01	0.26 − 05
0.1386	0.5325 + 00	0.13 − 03	0.2747 − 01	0.67 − 05
0.1848	0.4129 + 00	0.15 − 03	0.3279 − 01	0.11 − 04
0.2309	0.3239 + 00	0.16 − 03	0.3594 − 01	0.17 − 04
0.2771	0.2576 + 00	0.16 − 03	0.3758 − 01	0.22 − 04
0.3233	0.2081 + 00	0.16 − 03	0.3825 − 01	0.28 − 04
0.3695	0.1709 + 00	0.16 − 03	0.3839 − 01	0.35 − 04
0.4157	0.1427 + 00	0.16 − 03	0.3825 − 01	0.41 − 04
0.4619	0.1210 + 00	0.15 − 03	0.3801 − 01	0.47 − 04
0.5081	0.1041 + 00	0.15 − 03	0.3773 − 01	0.52 − 04
0.5543	0.9071 − 01	0.14 − 03	0.3743 − 01	0.57 − 04
0.6004	0.7977 − 01	0.14 − 03	0.3711 − 01	0.66 − 04
0.6466	0.7073 − 01	0.15 − 03	0.3678 − 01	0.78 − 04
0.6928	0.6315 − 01	0.15 − 03	0.3641 − 01	0.88 − 04
0.7390	0.5669 − 01	0.15 − 03	0.3601 − 01	0.98 − 04
0.7852	0.5111 − 01	0.15 − 03	0.3556 − 01	0.10 − 03
0.8314	0.4626 − 01	0.14 − 03	0.3505 − 01	0.10 − 03
0.8776	0.4194 − 01	0.14 − 03	0.3448 − 01	0.11 − 03
0.9238	0.3816 − 01	0.14 − 03	0.3388 − 01	0.12 − 03
0.9699	0.3481 − 01	0.14 − 03	0.3325 − 01	0.13 − 03
1.0161	0.3183 − 01	0.14 − 03	0.3261 − 01	0.14 − 03
1.0623	0.2919 − 01	0.14 − 03	0.3196 − 01	0.15 − 03
1.1085	0.2684 − 01	0.14 − 03	0.3132 − 01	0.15 − 03
1.1547	0.2474 − 01	0.13 − 03	0.3070 − 01	0.16 − 03
1.2009	0.2287 − 01	0.13 − 03	0.3010 − 01	0.17 − 03
1.2471	0.2118 − 01	0.13 − 03	0.2950 − 01	0.18 − 03
1.2933	0.1966 − 01	0.13 − 03	0.2891 − 01	0.19 − 03
1.3395	0.1829 − 01	0.13 − 03	0.2835 − 01	0.19 − 03
1.3856	0.1704 − 01	0.13 − 03	0.2779 − 01	0.20 − 03
1.4318	0.1590 − 01	0.12 − 03	0.2725 − 01	0.20 − 03
1.4780	0.1489 − 01	0.11 − 03	0.2676 − 01	0.20 − 03
1.5242	0.1399 − 01	0.10 − 03	0.2633 − 01	0.20 − 03
1.5704	0.1317 − 01	0.10 − 03	0.2593 − 01	0.20 − 03
1.6166	0.1242 − 01	0.10 − 03	0.2553 − 01	0.20 − 03
1.6628	0.1172 − 01	0.99 − 04	0.2513 − 01	0.21 − 03
1.7090	0.1107 − 01	0.97 − 04	0.2475 − 01	0.21 − 03
1.7551	0.1049 − 01	0.98 − 04	0.2440 − 01	0.22 − 03
1.8013	0.9951 − 02	0.10 − 03	0.2405 − 01	0.24 − 03
1.8475	0.9446 − 02	0.10 − 03	0.2372 − 01	0.26 − 03
1.8937	0.8974 − 02	0.11 − 03	0.2338 − 01	0.29 − 03

[a] The second column gives the values of $f(t) = \langle \mathbf{v}(t)\,\mathbf{v}(0)\rangle / \langle \mathbf{v}(0)\,\mathbf{v}(0)\rangle$ calculated from 20 runs of 20,000 time steps. The third column gives the standard error σ_s on $f(t)$. The fourth and fifth columns give the same results for the function $f(t)\,t^{3/2}$.

Table II. *(Continued)*

t	$f(t)$	σ_s	$f(t)\,t^{3/2}$	σ_s
1.9399	0.8536 − 02	0.12 − 03	0.2306 − 01	0.31 − 03
1.9861	0.8123 − 02	0.12 − 03	0.2273 − 01	0.32 − 03
2.0323	0.7751 − 02	0.11 − 03	0.2245 − 01	0.32 − 03
2.0785	0.7430 − 02	0.11 − 03	0.2226 − 01	0.32 − 03
2.1246	0.7130 − 02	0.10 − 03	0.2208 − 01	0.32 − 03
2.1708	0.6825 − 02	0.10 − 03	0.2183 − 01	0.32 − 03
2.2170	0.6522 − 02	0.10 − 03	0.2153 − 01	0.33 − 03
2.2632	0.6238 − 02	0.10 − 03	0.2124 − 01	0.35 − 03
2.3094	0.5977 − 02	0.11 − 03	0.2097 − 01	0.38 − 03
2.3556	0.5735 − 02	0.12 − 03	0.2073 − 01	0.41 − 03
2.4018	0.5530 − 02	0.12 − 03	0.2058 − 01	0.44 − 03
2.4480	0.5357 − 02	0.12 − 03	0.2052 − 01	0.45 − 03
2.4942	0.5202 − 02	0.12 − 03	0.2049 − 01	0.45 − 03
2.5403	0.5047 − 02	0.12 − 03	0.2043 − 01	0.46 − 03
2.5865	0.4870 − 02	0.11 − 03	0.2026 − 01	0.47 − 03
2.6327	0.4680 − 02	0.11 − 03	0.1999 − 01	0.46 − 03
2.6789	0.4515 − 02	0.10 − 03	0.1979 − 01	0.45 − 03
2.7251	0.4368 − 02	0.10 − 03	0.1965 − 01	0.45 − 03
2.7713	0.4234 − 02	0.10 − 03	0.1953 − 01	0.46 − 03
2.8175	0.4114 − 02	0.97 − 04	0.1946 − 01	0.46 − 03
2.8637	0.4000 − 02	0.91 − 04	0.1938 − 01	0.44 − 03
2.9098	0.3882 − 02	0.84 − 04	0.1927 − 01	0.42 − 03
2.9560	0.3753 − 02	0.80 − 04	0.1907 − 01	0.41 − 03
3.0022	0.3612 − 02	0.76 − 04	0.1879 − 01	0.39 − 03
3.0484	0.3471 − 02	0.71 − 04	0.1847 − 01	0.37 − 03
3.0946	0.3344 − 02	0.68 − 04	0.1820 − 01	0.37 − 03
3.1408	0.3244 − 02	0.67 − 04	0.1805 − 01	0.37 − 03
3.1870	0.3182 − 02	0.69 − 04	0.1810 − 01	0.39 − 03
3.2332	0.3139 − 02	0.78 − 04	0.1825 − 01	0.45 − 03
3.2793	0.3098 − 02	0.90 − 04	0.1840 − 01	0.53 − 03
3.3255	0.3054 − 02	0.10 − 03	0.1852 − 01	0.61 − 03
3.3717	0.3007 − 02	0.11 − 03	0.1862 − 01	0.68 − 03
3.4179	0.2960 − 02	0.12 − 03	0.1870 − 01	0.74 − 03
3.4671	0.2903 − 02	0.12 − 03	0.1871 − 01	0.77 − 03
3.5103	0.2838 − 02	0.12 − 03	0.1866 − 01	0.78 − 03
3.5565	0.2758 − 02	0.12 − 03	0.1849 − 01	0.78 − 03
3.6027	0.2674 − 02	0.11 − 03	0.1829 − 01	0.76 − 03
3.6489	0.2598 − 02	0.10 − 03	0.1811 − 01	0.75 − 03

3. MICROCANONICAL MONTE CARLO METHODS

The purpose of this section is to provide a test of the equivalence of time averages and microcanonical averages as far as equilibrium properties are concerned. In order to perform this comparison, we have adapted two different MC methods, one proposed by Ray,(8) the other by Creutz.(9)

Let us proceed with the first method. Let $W_E(\mathbf{R})$ be the microcanonical probability density of a system of N atoms, E being the total energy of that system and $\mathbf{R} = \{\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_N\}$. As shown by Ray, this probability can be written, for large values of N, in the form

$$W_E(\mathbf{R}) = C[E - U(\mathbf{R})]^{3N/2-1}\,\Theta(E - U(\mathbf{R})) \tag{5}$$

where $U(\mathbf{R})$ is the potential energy of the N atoms, C is a constant, and Θ is the Heaviside function, which guarantees that the total kinetic energy of the system, $K(\mathbf{R}) = E - U(\mathbf{R})$, stays positive. When, in order to sample phase space, a random displacement of an atom is carried out resulting in a new configuration $\mathbf{R}'$, the potential energy is changed by an amount $\Delta U(\mathbf{R}, \mathbf{R}')$. The probability ratio $P_E(\mathbf{R}, \mathbf{R}')$ of the two configurations defined by $\mathbf{R}$ and $\mathbf{R}'$ is then

$$P_E(\mathbf{R}, \mathbf{R}') = \left[\frac{E - U(\mathbf{R}')}{E - U(\mathbf{R})}\right]^{3N/2-1} \tag{6}$$

whence

$$\log P_E(\mathbf{R}, \mathbf{R}') = \left(\frac{3N}{2} - 1\right)\left[\frac{-\Delta U(\mathbf{R}, \mathbf{R}')}{E - U(\mathbf{R}, \mathbf{R}')}\right] + O\left(\frac{1}{N}\right) \tag{7}$$

If we define a "configuration temperature" through the equation

$$T(\mathbf{R}) = \frac{2(E - U(\mathbf{R}))}{3Nk_\mathrm{B}} = \frac{2K(\mathbf{R})}{3Nk_\mathrm{B}} \tag{8}$$

we obtain for $P_E(\mathbf{R}, \mathbf{R}')$ the following expression, which is approximate, but very accurate when N is large:

$$P_E(\mathbf{R}, \mathbf{R}') \cong \exp[-\Delta U(\mathbf{R}, \mathbf{R}')/k_\mathrm{B}T(\mathbf{R})] \tag{9}$$

Our microcanonical MC procedure is now obvious. It proceeds according to the Metropolis method, except that, instead of having a fixed temperature, one uses a configuration temperature which is calculated anew at each step. More specifically, the following procedure is used: starting with an initial configuration $\mathbf{R}$ and a total energy E, we compute $U(\mathbf{R})$, which leads to the configuration temperature $T(\mathbf{R})$. Then we try to make a random displacement of one of the atoms, leading to the new configuration $\mathbf{R}'$. Then $\Delta U(\mathbf{R}, \mathbf{R}')$ is calculated. From there, a new kinetic energy is found according to $K(\mathbf{R}') = K(\mathbf{R}) - \Delta U(\mathbf{R}, \mathbf{R}')$, so that the total energy is conserved. The move is rejected if this quantity ever happens to be negative [but unless the temperature is very low or the random move unreasonably large, this should not occur, because $K(\mathbf{R})$ is of the order N and $\Delta U(\mathbf{R}, \mathbf{R}')$ is of order 1]. In the opposite case, it is accepted or rejected according to the probability $\min[1, P_E(\mathbf{R}, \mathbf{R}')]$. We then iterate, starting either from the new configuration if it has been accepted or otherwise from the old one. The only difference with the standard Metropolis method is that, instead of being fixed once and for all, the temperature is calculated at each step, so as to keep the total energy constant.

The second MC method considered here for the microcanonical ensemble is an extension of that proposed by Creutz for systems with no kinetic energy. In the Creutz method, there is a demon with energy E_D in charge carrying energy from one atom to the other while the total energy of the system is kept constant. The difference is that here both potential and kinetic energy must be changed. The total energy may thus be written as

$$E = \tfrac{1}{2} \sum_i m\mathbf{v}_i^2 + U(\mathbf{R}) + E_D = E_S(\mathbf{V}, \mathbf{R}) + E_D \tag{10}$$

where $\mathbf{v}_i$ is the velocity vector of the ith atom and $\mathbf{V} = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_N\}$. In order to provide a microcanonical sampling including precisely the same constraints as in the MD simulations, the demon may also be given a momentum $\mathbf{p}_D$ so as to conserve the total momentum $\mathbf{P}$,

$$\mathbf{P} = \sum_i \mathbf{p}_i + \mathbf{p}_D \tag{11}$$

which is chosen equal to zero. The demon is allowed to exchange energy and momentum with the atoms, but subjected to the conditions that E and $\mathbf{P}$ are kept constant. The sampling procedure is carried out in the following way. Starting from an initial condition with velocities $\mathbf{V}$ and positions $\mathbf{R}$, and where the demon energy has some definite value E_D (which may be taken as equal to zero at the very beginning of the computation), we change the velocity $\mathbf{v}_i$ and the position $\mathbf{r}_i$ by a random amount: $\mathbf{v}_i$ and $\mathbf{r}_i$ become $\mathbf{v}_i'$ and $\mathbf{r}_i'$, respectively. This random move in phase space modifies $E_S(\mathbf{V}, \mathbf{R})$ by the amount $\Delta E_S = E_S(\mathbf{V}', \mathbf{R}') - E_S(\mathbf{V}, \mathbf{R})$. It will be accepted if the three conditions

$$\begin{aligned} m(v_i'^x - v_i^x)\, p_D^x &< 0 \\ m(v_i'^y - v_i^y)\, p_D^y &< 0 \\ m(v_i'^z - v_i^z)\, p_D^z &< 0 \end{aligned} \tag{12}$$

and either one of the two following conditions are realized

$$\begin{aligned} \Delta E_S &< 0 \\ E_D - \Delta E_S &> 0 \end{aligned} \tag{13}$$

When the move is accepted, E_D is replaced by $E_D - \Delta E_S$ and $\mathbf{p}_D$ is replaced by $\mathbf{p}_D + m(\mathbf{v}_i' - \mathbf{v}_i)$. As is explained by Creutz,$^{(9)}$ statistical mechanics implies that the probability distribution of E_D is a canonical distribution, i.e.,

$$P(E_D) \cong \exp(-E_D/k_B T) \tag{14}$$

This also implies that $\langle E_D \rangle = k_B T$, so that E_D is of order 1. when E_S is of order N. Due to the random sampling of velocities, this latter method is less efficient than the preceding one, by a factor of around 8.

The two procedures were applied to the system of 4000 atoms described in the previous section with a total energy of $\sim 3.74\varepsilon$ per atom. The thermodynamic properties calculated by these microcanonical MC methods are summarized in Table I. As expected they are identical within statistical uncertainties ($\sim 1.0\%$) with the results of time-reversible MD simulations.

4. CONCLUSION

Based on computer simulations of classical fluids, the present paper provides a realistic model illustrating the foundations of the Boltzmann's approach to statistical mechanics. Paradoxically, computer simulations have sometimes been used against Boltzmann's point of view, due to their imperfections, which we show here how to eradicate. Our computer model is realistic enough to incorporate what seems to be the main features of real liquids, and indeed shows the observed irreversible behavior of such systems despite the fact that it is deterministic, time reversible, and not subject to small external random perturbations.

APPENDIX

We give below the essential parts of the Fortran program which was used to perform explicit time-reversible MD simulations. To take into account the short range of the interaction, we use tables of neighboring atoms which are updated at regular time intervals, a procedure which does not preclude the generation of time-reversible trajectories. The present form of the program supposes that the computer is able to perform integer arithmetic operations on integers of 64 bits.

Program Fortran

```
C IM is the number of atoms.
C ITM is the number of integration steps before time reversal.
C RO is the density.
C EL is the side of the cubic volume with periodic boundary conditions.
C SC is the square of the range of the repulsive part of the Lennard-Jones
  pair potential.
C R2M is the square of the radius of the sphere enclosing the neighbors of
  a given atom.
C H is the integration time step.
C The number of neighbors of each atom is in the array IK.
C The indices of the neighbors are in the array INW.
C The array INW is updated after ITMOD, 2*ITMOD,..., integration
  steps.
C Array XP floating coordinates of the atoms at time t = 0.
C Array XPP floating coordinates of the atoms at time t = -h.
C Array INIT integer coordinates of the atoms at time t = 0.
C Array IXP integer coordinates of the atoms at time t.
C Array IXPP integer coordinates of the atoms at time t - h.
C Array IX integer coordinates of the atoms at time t + h.
      PI = ACOS(-1.)
      EL = (IM/RO)**(1./3.)
      CST = 48.*H**2
      NP = 60
      ICT = 2**NP
      ICT2 = 2**(NP - 1)
      SC = 2.**(1./3.)
      R2M = (SQRT(SC) + H*20.*V0*ITMOD)**2
C The initial configuration XP is an FCC lattice.
C The subroutine function HAS generates uniform random numbers
  between ]0,1.].
C The coordinates XPP differs from XP by Gaussian random displacements:
C (array DIS) of variance V0**2/2. and average UZ.
      DO 9 IC = 1,3
      UZ(IC) = 0.
      DO 8 I = 1,IM
      Z1 = HAS(NAS)
      Z2 = HAS(NAS)
      DIS(I,IC) = H*V0*SQRT(-ALOG(Z1))*COS(2.*PI*Z2)
      UZ(IC) = UZ(IC) + DIS(I,IC)
  8   CONTINUE
      UZ(IC) = UZ(IC)/IM
```

```
  9  CONTINUE
     DO 10 IC = 1,3
     DO 10 I = 1,IM
     IXP(I,IC) = INT(XP(I,IC)*ICT/EL)
     INIT(I,IC) = IXP(I,IC)
     XPP(I,IC) = XP(I,IC) - DIS(I,IC) - UZ(IC)
     IXPP(I,IC) = INT(XPP(I,IC)*ICT/EL)
 10  CONTINUE
     IT = 0
     IYY = 0
     NTD = 0
     IV = 1
C Beginning of the integration loop.
 100 IT = IT + IV
     DO 300 IC = 1,3
     DO 300 I = 1,IM
     IFP(I,IC) = 0
 300 CONTINUE
     IF(MOD(NTD,ITMOD).EQ.0) THEN
C Computation of the table INW of neighboring atoms.
     DO 49 I = 1,IM
     IK(I) = 0
 49  CONTINUE
     DO 29 I = 1,IM - 1
     NN = IK(I)
     DO 19 J = I + 1,IM
    ;  ICC(J,1) = ICC(J,1) - ICT
     IF((ICC(J,1).LT.0).AND.(ABS(ICC(J,1)).GT.ICT2))
    ;  ICC(J,1) = ICC(J,1) + ICT
     IF((ICC(J,1).LT.0).AND.(ABS(ICC(J,1)).GT.ICT2))
       ICC(J,1) = ICC(J,1) + ICT
     CC(J,1) = ICC(J,1)*EL/ICT
     ... idem for the other two coordinates ... .
     R2(J) = (CC(J,1)**2 + CC(J,2)**2 + CC(J,3)**2)
 19  CONTINUE
     DO 30 J = I + 1,IM
     IF(R2(J).GT.R2M) GO TO 30
     NN = NN + 1
     IK(J) = IK(J) + 1
     INW(J,IK(J)) = I
     INW(I,NN) = J
 30  CONTINUE
     IK(I) = NN
```

```
  29  CONTINUE
      ENDIF
      NTD = NTD + 1
C Computation of the distances and forces IFP.
      DO 32 I = 1,IM
      IF(IK(I).NE.0) THEN
      DO 42 J = 1,IK(I)
      IP = INW(I,J)
      ICC(J,1) = IXP(I,1) - IXP(IP,1)
      IF((ICC(J,1).GT.0).AND.(ABS(ICC(J,1)).GT.ICT2))
     ;  ICC(J,1) = ICC(J,1) - ICT
      IF((ICC(J,1).LT.0).AND.(ABS(ICC(J,1)).GT.ICT2))
     ;  ICC(J,1) = ICC(J,1) + ICT
      CC(J,1) = ICC(J,1)*EL/ICT
      ... idem for the other two coordinates ... .
      R2(J) = (CC(J,1)**2 + CC(J,2)**2 + CC(J,3)**2)
      PH(J) = 0.
      IF(R2(J).LE.SC) PH(J) = CST*(1./R2(J)**7 - 0.5/R2(J)**4)
  42  CONTINUE
      DO 40 IP = 1,IK(I)
      IPH(IP,1) = INT(ICT*PH(IP)*CC(IP,1)/EL)
      IPH(IP,2) = INT(ICT*PH(IP)*CC(IP,2)/EL)
      IPH(IP,3) = INT(ICT*PH(IP)*CC(IP,3)/EL)
  40  CONTINUE
      DO 20 IP = 1,IK(I)
      IFP(I,1) = IFP(I,1) + IPH(IP,1)
      IFP(I,2) = IFP(I,2) + IPH(IP,2)
      IFP(I,3) = IFP(I,3) + IPH(IP,3)
  20  CONTINUE
      ENDIF
  32  CONTINUE
C Integration of the motion (IFP included the factor h**2).
C The coordinates are modified according to the periodic boundary
  conditions.
      DO 50 IC = 1,3
      DO 50 I = 1,IM
      IX(I,IC) = 2*IXP(I,IC) - IXPP(I,IC) + IFP(I,IC)
      IF(IX(I,IC).GT.ICT) THEN
      IX(I,IC) = IX(I,IC) - ICT
      IXP(I,IC) = IXP(I,IC) - ICT
      IXPP(I,IC) = IXPP(I,IC) - ICT
      ELSE
      IF(IX(I,IC).LT.0) THEN
```

```
      IX(I,IC)=IX(I,IC)+ICT
      IXP(I,IC)=IXP(I,IC)+ICT
      IXPP(I,IC)=IXPP(I,IC)+ICT
      ENDIF
      ENDIF
   50 CONTINUE
      IF(IT.EQ.ITM) THEN
      IF(IYY.EQ.0) THEN
C End of the forward trajectory.
C Beginning of the backward trajectory.
      DO 66 IC=1,3
      DO 66 I=1,IM
      ITEMP(I,IC)=IXPP(I,IC)
      IXPP(I,IC)=IX(I,IC)
      IX(I,IC)=ITEMP(I,IC)
   66 CONTINUE
      DO 56 IC=1,3
      DO 56 I=1,IM
      IF(IXP(I,IC).GT.ICT) THEN
      IX(I,IC)=IX(I,IC)-ICT
      IXP(I,IC)=IXP(I,IC)-ICT
      IXPP(I,IC)=IXPP(I,IC)-ICT
      ELSE
      IF(IXP(I,IC).LT.0) THEN
      IX(I,IC)=IX(I,IC)+ICT
      IXP(I,IC)=IXP(I,IC)+ICT
      IXPP(I,IC)=IXPP(I,IC)+ICT
      ENDIF
      ENDIF
   56 CONTINUE
      IV= -IV
      IT=IT-IV
      ITM=1
      IYY=1
      ELSE
C The atoms have gone back to their initial positions.
      IP=0
      DO 71 IC=1,3
      DO 71 I=1,IM
      IP=IP+ABS(IXP(I,IC)-INIT(I,IC))
   71 CONTINUE
      WRITE (21,*)'IP',IP
```

```
C End of the backward trajectory.
      STOP
      ENDIF
      ELSE
C Time evolution of one step.
      DO 60 IC = 1,3
      DO 60 I = 1,IM
      IXPP(I,IC) = IXP(I,IC)
      IXP(I,IC) = IX(I,IC)
   60 CONTINUE
      ENDIF
      GO TO 100
      END
```

ACKNOWLEDGMENT

We thank J. Lebowitz for stimulating and helpful discussions.

REFERENCES

1. J. L. Lebowitz, Lecture presented at the Conference on Physical Origin of Time Asymmetry, Mazagon, Spain (1991).
2. K. Popper, *Unended Quest* (Collins, Glasgow, 1978).
3. I. Prigogine and I. Stengers, *Entre le temps et l'éternité* (Fayard, Paris, 1988).
4. J. Orban and A. Bellemans, *Phys. Lett.* **224**:620 (1967).
5. L. Verlet, *Phys. Rev.* **159**:98 (1967).
6. D. Levesque and W. T. Ashurst, *Phys. Rev. Lett.* **33**:277 (1974).
7. J. J. Erpenbeck and W. W. Wood, *Phys. Rev. A* **26**:1648 (1982); **32**:412 (1985); J. J. Erpenbeck, *Phys. Rev. A* **35**:218 (1987).
8. J. R. Ray, *Phys. Rev. A* **44**:4061 (1991).
9. H. Creutz, *Phys. Rev. Lett.* **50**:1411 (1983).
10. N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller, *J. Chem. Phys.* **21**:1087 (1953).
11. A. Rahman, *Phys. Rev.* **136**:405 (1964).
12. J. Delambre, *Mem. Acad. Turin* **5**:143 (1790–1793).
13. W. E. Milne, *Numerical Solution of Differential Equations* (Wiley, New York, 1960).
14. C. Störmer, *Arch. Sci. Phys. Nat. Genève* (1907).
15. C. Störmer, in *Congress International Mathematiques Strasbourg* (Toulouse, 1921), p. 24.
16. W. G. Hoover, *Molecular Dynamics* (Springer, 1986), p. 3.