On the calculation of velocity-dependent properties in molecular dynamics simulations using the leapfrog integration algorithm

Michel A. Cuendet and Wilfred F. van Gunsteren

Citation: J. Chem. Phys. **127**, 184102 (2007); doi: 10.1063/1.2779878
View online: http://dx.doi.org/10.1063/1.2779878
View Table of Contents: http://jcp.aip.org/resource/1/JCPSA6/v127/i18
Published by the American Institute of Physics.

On the calculation of velocity-dependent properties in molecular dynamics simulations using the leapfrog integration algorithm

Michel A. Cuendet
Laboratory of Physical Chemistry, Swiss Federal Institute of Technology ETH, 8093 Zürich, Switzerland and Laboratory of Computational Biochemistry and Chemistry, Swiss Federal Institute of Technology EPFL, 1015 Lausanne, Switzerland

Wilfred F. van Gunsteren
Laboratory of Physical Chemistry, Swiss Federal Institute of Technology ETH, 8093 Zürich, Switzerland

(Received 22 September 2006; accepted 14 August 2007; published online 8 November 2007)

Widely used programs for molecular dynamics simulation of (bio)molecular systems are the Verlet and leapfrog algorithms. In these algorithms, the particle velocities are less accurately propagated than the positions. Important quantities for the simulation such as the temperature and the pressure involve the squared velocities at full time steps. Here, we derive an expression for the squared particle velocity at full time step in the leapfrog scheme, which is more accurate than the standardly used one. In particular, this allows us to show that the full time step kinetic energy of a particle is more accurately computed as the average of the kinetic energies at previous and following half steps than as the square of the average velocity as implemented in various molecular dynamics codes. Use of the square of the average velocity introduces a systematic bias in the calculation of the instantaneous temperature and pressure of a molecular dynamics system. We show the consequences when the system is coupled to a thermostat and a barostat. © *2007 American Institute of Physics*. [DOI: 10.1063/1.2779878]

I. INTRODUCTION

Classical dynamics for a set of N particles or atoms in three dimensions is most generally formulated[1] as Hamilton's equations of motion for the generalized positions $q=(q_1,\ldots,q_{3N})$ and momenta $p=(p_1,\ldots,p_{3N})$,

$$\dot{q}_i = \frac{\partial H}{\partial p_i}(p,q), \tag{1}$$

$$\dot{p}_i = -\frac{\partial H}{\partial q_i}(p,q). \tag{2}$$

The Hamiltonian function describing conservative systems of particles with mass m_i and Cartesian coordinates $r=(r_1,\ldots,r_{3N})$ is

$$H(p,r) = \sum_{i=1}^{3N} \frac{p_i^2}{2m_i} + V(r), \tag{3}$$

i.e., the kinetic energy term only depends on the momenta or velocities $v_i=p_i/m_i$, and the potential energy term depends only on the positions r. Inserting Eq. (3) into Eqs. (1) and (2), Newton's equations of motion are obtained,

$$\dot{r}_i(t) = v_i(t), \tag{4}$$

$$\dot{v}_i(t) = \frac{f_i(t)}{m_i}, \tag{5}$$

$$f_i(t) = -\frac{\partial}{\partial r_i} V(r(t)). \tag{6}$$

Due to the separation of the p and q dependences in Eq. (3), Eqs. (4) and (5) are equivalent to a second-order differential equation,

$$\ddot{r}_i(t) = \frac{1}{m_i} f_i(r(t)), \tag{7}$$

in which no terms in $\dot{r}(t)$ are present. This implies that the computation of velocities is not required to solve Eq. (7). This equation also shows that Newton's equations of motion are invariant under time reversal. Both properties should be taken into account when selecting an algorithm to integrate Eqs. (4) and (5).

Numerical methods to solve sets of ordinary differential equations can be found in the literature.[2–4] They are based on finite differences and solve the equations by stepping forward in time using time steps Δt. Since the computation of the forces [Eq. (6)] is much more expensive than manipulation of positions and velocities, only integration schemes that involve only one force evaluation per time step are of use in molecular dynamics (MD) simulation. This rules out the generally very efficient Runge-Kutta algorithms. A second consideration when selecting an integration algorithm derives from the nature of the energy hypersurface $V(r)$ for molecular systems in the condensed phase. The function $V(r)$ is very rugged, i.e., it has a mountainous character with many valleys. Since a system often visits the bottoms of these valleys characterized by positive second derivatives of the potential energy $V(r)$ or negative first derivatives of the forces with

respect to r, an efficient MD algorithm should at least include a correct treatment of the first derivative of the forces in order to avoid systematic errors.[5] This means that the order of the algorithm, defined as the highest order of the time step Δt included in the equations for the propagation of the positions, should at least be three. In complex condensed phase molecular systems, it is not efficient to use an integration algorithm beyond third or fourth order[6] because the higher derivatives of the forces display a nonsystematic behavior.

These considerations (time reversibility, one force calculation per step, and up to third-order accuracy in Δt) have led to the widespread use of the Verlet[7–10] and leapfrog[11] algorithms in MD simulation. We note that these algorithms satisfy the condition of symplecticity,[12–15] i.e., they preserve the Poincaré invariants characteristic of continuous Hamiltonian systems. In addition, the resulting discrete trajectories represent the exact evolution of a "shadow" Hamiltonian system with a Hamiltonian function[16–18] arbitrarily close to the physical Hamiltonian, which hinders systematic energy drifts. The symplectic property of the Verlet and leapfrog integrators is most directly evidenced when deriving[15,19] them using a Trotter expansion of the Liouville evolution operator. This formalism also reveals[20,21] that both algorithms generate trajectories with equivalent positions but slightly different velocities. Indeed, as shown in the following, Verlet and leapfrog integrators both propagate the positions with an accuracy of third order in Δt, whereas the velocities are propagated to first and to second order, respectively.

Since most molecular properties of interest depend only on the particle positions, the relatively low accuracy of the velocity propagation is often not problematic. Yet, a higher-accuracy expression for the velocities would be welcome, e.g., when calculating kinetic energies, where inaccurate expressions may introduce a systematic bias. This is of concern when using the kinetic energies to couple the system to a heat bath in constant temperature simulations, or when the internal pressure of the system as calculated from the virial is regulated. Here we derive an expression for the particle velocities at full time steps, which is more accurate than the standard leapfrog expression.

In Sec. II, the relevant formulas of the Verlet and leapfrog integration algorithms are derived using the Taylor expansion approach. The more accurate expression for the squared velocities in the leapfrog scheme is derived in Sec. III and illustrated with simple numerical examples in Sec. IV. Section V contains a short discussion of the applicability of the proposed expressions.

II. THE VERLET AND LEAPFROG ALGORITHMS FOR MD SIMULATION

In order to simplify the notation, we consider in the following only Cartesian components x, v, and f of the position, velocity, and force of a single particle, respectively, and use the subindex n to indicate the time point t_n, i.e., $t_{n+1} \equiv t_n + \Delta t$. The Verlet integration scheme is most easily derived[5] by expanding the positions at time $t_n + \Delta t$ in a Taylor series in Δt at time t_n,

$$x_{n+1} = x_n + \dot{x}_n \Delta t + \frac{1}{2!}\ddot{x}_n (\Delta t)^2 + \frac{1}{3!}\dddot{x}_n (\Delta t)^3 + \mathcal{O}((\Delta t)^4). \quad (8)$$

Adding the corresponding expansion for $t_n - \Delta t$, rearranging terms, and using Eq. (7), one obtains the Verlet algorithm,

$$x_{n+1} = 2x_n - x_{n-1} + \frac{1}{m} f_n (\Delta t)^2 + \mathcal{O}((\Delta t)^4), \quad (9)$$

which indeed fulfills the three criteria for an efficient MD algorithm mentioned above. An expression for the velocity can be obtained by subtracting the Taylor expansion for $t_n - \Delta t$ from the one for $t_n + \Delta t$, rearranging, and using Eq. (7),

$$v_n = \frac{1}{2\Delta t}(x_{n+1} - x_{n-1}) - \frac{1}{6m}\dot{f}_n (\Delta t)^2 + \mathcal{O}((\Delta t)^4). \quad (10)$$

In the standard Verlet algorithm, the term containing $\dot{f}_n$ is omitted, leading to a first-order accuracy of the Verlet velocity propagation,

$$v_n = \frac{1}{2\Delta t}(x_{n+1} - x_{n-1}). \quad (11)$$

The accuracy can be enhanced by using a time-reversal invariant expression for $\dot{f}_n$,

$$\dot{f}_n = \frac{1}{2\Delta t}(f_{n+1} - f_{n-1}), \quad (12)$$

leading to

$$v_n = \frac{1}{2\Delta t}(x_{n+1} - x_{n-1}) - \frac{1}{12m}(f_{n+1} - f_{n-1})\Delta t + \mathcal{O}((\Delta t)^4). \quad (13)$$

Since the Verlet algorithm (9) to propagate the positions does not depend explicitly on velocities, a coupling of the molecular system to a heat bath or thermostat, which will influence the velocities, is not straightforward. Such a coupling is more easily implemented using the leapfrog integration algorithm.

The leapfrog integration scheme is most easily derived by expanding the positions or velocities at time $t_n + \Delta t/2$ in a Taylor series in $\Delta t/2$ at time point t. Subtracting the expansions of the positions for $\Delta t/2$ and $-\Delta t/2$ from each other, rearranging terms, using Eq. (7), and then shifting all time points by $\Delta t/2$, one obtains

$$x_{n+1} = x_n + v_{n+1/2}\Delta t + \frac{1}{24m}\dot{f}_{n+1/2}(\Delta t)^3 + \mathcal{O}((\Delta t)^5). \quad (14)$$

Using a time-reversal invariant expression for $f_{n+1/2}$,

$$\dot{f}_{n+1/2} = \frac{1}{\Delta t}(f_{n+1} - f_n), \quad (15)$$

one obtains

$$x_{n+1} = x_n + v_{n+1/2}\Delta t + \frac{1}{24m}(f_{n+1} - f_n)(\Delta t)^2 + \mathcal{O}((\Delta t)^5). \quad (16)$$

The expression for the velocity propagation is obtained by subtracting the expansions of the velocities for $\Delta t/2$ and

$-\Delta t/2$ from each other, rearranging terms, and using Eq. (7),

$$v_{n+1/2} = v_{n-1/2} + \frac{1}{m} f_n \Delta t + \frac{1}{24m} \ddot{f}_n (\Delta t)^3 + \mathcal{O}((\Delta t)^5). \quad (17)$$

Using a time-reversal invariant expression for $\ddot{f}_n$,

$$\ddot{f}_n = \frac{1}{(\Delta t)^2}(f_{n+1} - 2f_n + f_{n-1}), \quad (18)$$

one obtains

$$v_{n+1/2} = v_{n-1/2} + \frac{1}{m} f_n \Delta t + \frac{1}{24m}(f_{n+1} - 2f_n + f_{n-1})\Delta t + \mathcal{O}((\Delta t)^5). \quad (19)$$

The standard leapfrog algorithm,

$$v_{n+1/2} = v_{n-1/2} + \frac{1}{m} f_n \Delta t + \mathcal{O}((\Delta t)^3), \quad (20)$$

$$x_{n+1} = x_n + v_{n+1/2}\Delta t + \mathcal{O}((\Delta t)^3), \quad (21)$$

is obtained by omitting the terms containing $\dot{f}$ and $\ddot{f}$ in Eqs. (14) and (17), respectively.

The leapfrog algorithm [Eqs. (21) and (20)] is exactly equivalent[5] to the Verlet algorithm [Eqs. (9) and (11)]. This can be shown by writing the velocity $v_{n-1/2}$ in terms of positions: shift the time in Eq. (21) by $-\Delta t$ and solve for $v_{n-1/2}$, then substitute the resulting expression in Eq. (20) and the latter in Eq. (21). This leads to Eq. (9), the Verlet position formula. The Verlet velocity formula (11) is obtained by using Eq. (21) to express $v_{n-1/2}$ and $v_{n+1/2}$ in terms of positions and then inserting the result into

$$v_n = \tfrac{1}{2}(v_{n-1/2} + v_{n+1/2}), \quad (22)$$

which is the formula for the leapfrog velocity at full time steps.

A higher-order expression for v_n in the leapfrog scheme can be obtained by subtracting the Taylor expansion for the position at $t_n - \Delta t$ from the one at $t_n + \Delta t$ and substituting the result in Eq. (13),

$$v_n = \tfrac{1}{2}(v_{n-1/2} + v_{n+1/2}) - \frac{1}{16m}(f_{n+1} - f_{n-1})\Delta t + \mathcal{O}((\Delta t)^4). \quad (23)$$

Comparing this expression for v_n to the one in Eq. (3.45) in Ref. 5, it is observed that the latter contains (wrongly) a factor 1/12 instead of 1/16. The factor 1/12 is obtained when using Eq. (11) instead of the more accurate Eq. (13) in the derivation of Eq. (23). Due to the occurrence of f_{n+1} in Eq. (23), which is unknown at t_n, Eq. (23) is not of much practical use.

An alternative higher-order expression for v_n in the leapfrog scheme can be obtained by subtracting from x_{n+1} as given in Eq. (16) the analogous expression for x_{n-1},

$$x_{n+1} - x_{n-1} = (v_{n-1/2} + v_{n+1/2})\Delta t + \frac{1}{24m}(f_{n+1} - f_{n-1})(\Delta t)^2 + \mathcal{O}((\Delta t)^5). \quad (24)$$

Solving for $(f_{n+1} - f_{n-1})$ and substituting the result into Eq. (23) yields

$$v_n = (v_{n-1/2} + v_{n+1/2}) - \frac{3}{2\Delta t}(x_{n+1} - x_{n-1}) + \mathcal{O}((\Delta t)^4), \quad (25)$$

which can be used to obtain *a posteriori* higher-accuracy velocities from a standard leapfrog trajectory. Due to the presence of x_{n+1}, its use in generating a trajectory is not practical.

III. A SIMPLE HIGHER-ORDER FORMULA FOR SQUARED VELOCITIES IN THE LEAPFROG SCHEME

A first expression for the squared velocity at time t_n can be obtained by taking the square of Eq. (23) and omitting terms of order $(\Delta t)^2$ and higher,

$$v_n^2 = \tfrac{1}{4}(v_{n-1/2}^2 + 2v_{n-1/2}v_{n+1/2} + v_{n+1/2}^2) - \frac{1}{16m}(v_{n-1/2} + v_{n+1/2})(f_{n+1} - f_{n-1})\Delta t + \mathcal{O}((\Delta t)^2). \quad (26)$$

In order to obtain an expression for $v_{n-1/2}v_{n+1/2}$ in terms of squared velocities and forces, we multiply Eq. (17) by $v_{n-1/2}$, multiply the time-inverted equivalent of Eq. (17) by $v_{n+1/2}$, and add the resulting equations, thereby omitting terms of order $(\Delta t)^3$ and higher,

$$2v_{n-1/2}v_{n+1/2} = v_{n-1/2}^2 + v_{n+1/2}^2 - \frac{1}{m}(v_{n+1/2} - v_{n-1/2})f_n \Delta t + \mathcal{O}((\Delta t)^3). \quad (27)$$

Inserting this result into Eq. (26) gives an alternative higher-order expression for v_n^2,

$$v_n^2 = v_n^2(\mathrm{I}) + v_n^2(\mathrm{II}) + v_n^2(\mathrm{III}) + \mathcal{O}((\Delta t)^2), \quad (28)$$

with the three terms I, II, and III defined, respectively, as

$$v_n^2(\mathrm{I}) = \tfrac{1}{2}(v_{n-1/2}^2 + v_{n+1/2}^2), \quad (29)$$

$$v_n^2(\mathrm{II}) = -\frac{1}{4m}(v_{n+1/2} - v_{n-1/2})f_n \Delta t, \quad (30)$$

$$v_n^2(\mathrm{III}) = -\frac{1}{16m}(v_{n-1/2} + v_{n+1/2})(f_{n+1} - f_{n-1})\Delta t. \quad (31)$$

The above derivation shows that expression (28) rigorously includes all terms up to order one in Δt. Expression (28) is therefore inherently closer than lower-order ones to the exact continuous squared velocity at time t_n. Term I is always positive. In order to estimate the signs of terms II and III, we replace finite difference expressions for v and f by their equivalent averages or derivatives. Term II represents the squared forces,

$$v_n^2(\mathrm{II}) \approx -\frac{1}{4m^2} f_n^2 (\Delta t)^2 \leqslant 0, \tag{32}$$

and term III expresses the correlation between the velocity and the force variation,

$$v_n^2(\mathrm{III}) \approx -\frac{1}{4m} v_n \dot{f}_n (\Delta t)^2. \tag{33}$$

Using the relations $\dot{f}=(\partial f/\partial x)(\mathrm{d}x/\mathrm{d}t)$ and $f=-\partial V/\partial x$, we find for the last term,

$$v_n^2(\mathrm{III}) \approx +\frac{1}{4m} v_n^2 \frac{\partial^2 V}{\partial x_n^2} (\Delta t)^2, \tag{34}$$

which will be positive for a convex ($\partial^2 V/\partial x^2 > 0$) potential energy surface. As noted in the Introduction, a MD system in the condensed phase spends most time in the bottoms of the energy hypersurface valleys, which are characterized by positive second derivatives. This implies that $v_n^2(\mathrm{III})$ is generally positive.

In the standard leapfrog (LF) scheme, the kinetic energy at full time step is evaluated using the average velocity at t_n, which by using Eq. (27) is equal to

$$\begin{aligned} v_n^2(\mathrm{LF}) &= \tfrac{1}{4}(v_{n-1/2} + v_{n+1/2})^2 \\ &= \tfrac{1}{2}(v_{n-1/2}^2 + v_{n+1/2}^2) - \frac{1}{4m}(v_{n+1/2} - v_{n-1/2}) f_n \Delta t \\ &= v_n^2(\mathrm{I}) + v_n^2(\mathrm{II}). \end{aligned} \tag{35}$$

The terms I and II are of orders $(\Delta t)^0$ and $(\Delta t)^1$, respectively. Compared to the correct higher-order expression (28), term III, which is also of order $(\Delta t)^1$, is missing. Because terms II and III generally have opposite signs, a systematic negative bias is associated with $v_n^2(\mathrm{LF})$, which yields a slightly too low kinetic energy. In order to estimate v_n^2, it is consequently more accurate to only use term I, which is simply the average (AVG) of the squared velocities at times $n-1/2$ and $n+1/2$,

$$v_n^2(\mathrm{AVG}) = v_n^2(\mathrm{I}) = \tfrac{1}{2}(v_{n-1/2}^2 + v_{n+1/2}^2), \tag{36}$$

and is unbiased at order $(\Delta t)^0$.

Computing the squared velocities with the full higher-order (HO) expression,

$$v_n^2(\mathrm{HO}) = v_n^2(\mathrm{I}) + v_n^2(\mathrm{II}) + v_n^2(\mathrm{III}) \tag{37}$$

is impractical because term III contains f_{n+1} which is not available at step n. An alternative is to keep both terms of order $(\Delta t)^1$, at the cost of abandoning the time reversibility in the third term of Eq. (28), by approximating $(f_{n+1}-f_{n-1})$ with $2(f_n-f_{n-1})$. This gives the higher-order nonreversible (HOnr) estimate,

$$\begin{aligned} v_n^2(\mathrm{HOnr}) &= \tfrac{1}{2}(v_{n-1/2}^2 + v_{n+1/2}^2) - \frac{1}{4m}(v_{n+1/2} - v_{n-1/2}) f_n \Delta t \\ &\quad - \frac{1}{8m}(v_{n-1/2} + v_{n+1/2})(f_n - f_{n-1}) \Delta t \\ &\quad + \mathcal{O}((\Delta t)^2), \end{aligned} \tag{38}$$

in which the third term will be denoted as $v_n^2(\mathrm{III}')$.

IV. APPLICATIONS

A. One-dimensional systems: Harmonic oscillator and Lennard-Jones particle pair

The effects of the different approximations discussed in the previous section can be illustrated by using a one-dimensional harmonic oscillator with force constant $m\omega^2$. An exact trajectory is given by $x(t)=\cos(\omega t)$. The amplitude, which depends on the initial conditions, is set to unity. The exact squared velocity at time t_n is then

$$v_n^2 = \omega^2 \sin^2(\omega t_n), \tag{39}$$

and the three terms of Eq. (28) are

$$\begin{aligned} v_n^2(\mathrm{I}) &= \tfrac{1}{2}\omega^2[\sin^2(\omega t_{n-1/2}) + \sin^2(\omega t_{n+1/2})] \\ &= \tfrac{1}{2}\omega^2[1 - \cos(2\omega t_n)\cos(\omega \Delta t)], \end{aligned} \tag{40}$$

$$\begin{aligned} v_n^2(\mathrm{II}) &= -\tfrac{1}{4}\omega^3 \Delta t[\sin(\omega t_{n+1/2}) - \sin(\omega t_{n-1/2})]\cos(\omega t_n) \\ &= -\tfrac{1}{4}\omega^3 \Delta t[1 + \cos(2\omega t_n)]\sin\left(\omega \frac{\Delta t}{2}\right), \end{aligned} \tag{41}$$

$$\begin{aligned} v_n^2(\mathrm{III}) &= -\tfrac{1}{16}\omega^3 \Delta t[\sin(\omega t_{n-1/2}) + \sin(\omega t_{n+1/2})][\cos(\omega t_{n+1}) \\ &\quad - \cos(\omega t_{n-1})] \\ &= +\tfrac{1}{8}\omega^3 \Delta t[1 - \cos(2\omega t_n)]\cos\left(\omega \frac{\Delta t}{2}\right)\sin(\omega \Delta t). \end{aligned} \tag{42}$$

Using Taylor series expansions around zero for the cosine and sine functions with arguments containing Δt, we find

$$\begin{aligned} v_n^2(\mathrm{I}) &= \omega^2 \sin^2(\omega t_n) + \tfrac{1}{4}\omega^2 \cos(2\omega t_n) \\ &\quad \times\left[(\omega \Delta t)^2 - \tfrac{1}{12}(\omega \Delta t)^4\right] + \mathcal{O}((\Delta t)^6), \end{aligned} \tag{43}$$

$$v_n^2(\mathrm{II}) = -\tfrac{1}{4}\omega^2 \cos^2(\omega t_n)\left[(\omega \Delta t)^2 - \tfrac{1}{24}(\omega \Delta t)^4\right] + \mathcal{O}((\Delta t)^6), \tag{44}$$

$$v_n^2(\mathrm{III}) = \tfrac{1}{4}\omega^2 \sin^2(\omega t_n)\left[(\omega \Delta t)^2 - \tfrac{7}{24}(\omega \Delta t)^4\right] + \mathcal{O}((\Delta t)^6). \tag{45}$$

Summing up these terms yields

$$\begin{aligned} v_n^2(\mathrm{HO}) &= \omega^2 \sin^2(\omega t_n) - \omega^2\left[\tfrac{1}{96} + \tfrac{1}{24}\sin^2(\omega t_n)\right](\omega \Delta t)^4 \\ &\quad + \mathcal{O}((\Delta t)^6), \end{aligned} \tag{46}$$

which shows that $v_n^2(\mathrm{HO})$ matches the exact squared velocity up to order $(\Delta t)^4$. For the approximation $v_n^2(\mathrm{III}')$ we find

$$\begin{aligned} v_n^2(\mathrm{III}') &= -\tfrac{1}{4}\omega^3 \Delta t \sin(\omega t_n)\cos\left(\omega \frac{\Delta t}{2}\right) \\ &\quad \times[\cos(\omega t_n)(1 - \cos(\omega \Delta t)) - \sin(\omega t_n)\sin(\omega \Delta t)], \end{aligned} \tag{47}$$

or using Taylor series expansions around zero for the cosine and sine functions with arguments containing Δt,

$$v_n^2(\mathrm{III}') = +\tfrac{1}{4}\omega^2 \sin^2(\omega t_n)\left[(\omega\Delta t)^2 - \tfrac{1}{2}\cot(\omega t_n)(\omega\Delta t)^3 - \tfrac{7}{24}(\omega\Delta t)^4 + \tfrac{5}{48}\cot(\omega t_n)(\omega\Delta t)^5\right] + \mathcal{O}((\Delta t)^6). \quad (48)$$

Summing up the terms $v_n^2(\mathrm{I})$, $v_n^2(\mathrm{II})$, and $v_n^2(\mathrm{III}')$ yields

$$v_n^2(\mathrm{HOnr}) = \omega^2 \sin^2(\omega t_n) - \tfrac{1}{16}\omega^2 \sin(2\omega t_n)(\omega\Delta t)^3 - \tfrac{1}{32}\omega^2\left[1 - \frac{2}{3}\cos(2\omega t_n)\right](\omega\Delta t)^4 + \tfrac{5}{384}\omega^2 \sin(2\omega t_n)(\omega\Delta t)^5 + \mathcal{O}((\Delta t)^6), \quad (49)$$

which shows that $v_n^2(\mathrm{HOnr})$ matches the exact squared velocity up to order $(\Delta t)^3$.

We define the absolute error of estimator XX as $\Delta v_n^2(\mathrm{XX}) = v_n^2(\mathrm{XX}) - v_n^2$, where v_n^2 stands for the exact squared velocity at step n. The time averaged absolute error or bias of an estimator is then denoted as $\langle \Delta v_n^2(\mathrm{XX})\rangle$. Using Eqs. (40)–(42) and (48), we can compute the biases of the various v_n^2 estimators defined in Eqs. (36), (35), (37), and (38), respectively,

$$\langle \Delta v_n^2(\mathrm{AVG})\rangle = 0, \quad (50)$$

$$\langle \Delta v_n^2(\mathrm{LF})\rangle = -\tfrac{1}{4}\omega^3 \Delta t \sin\left(\omega\frac{\Delta t}{2}\right) = -\tfrac{1}{8}\omega^4(\Delta t)^2 + \tfrac{1}{192}\omega^6(\Delta t)^4 + \mathcal{O}((\Delta t)^6), \quad (51)$$

$$\langle \Delta v_n^2(\mathrm{HO})\rangle = -\tfrac{1}{4}\omega^3 \Delta t \sin^3\left(\omega\frac{\Delta t}{2}\right) = -\tfrac{1}{32}\omega^6(\Delta t)^4 + \mathcal{O}((\Delta t)^6) = \langle \Delta v_n^2(\mathrm{HOnr})\rangle. \quad (52)$$

This shows that the leapfrog estimator has a systematic bias of order $(\Delta t)^2$, whereas $v_n^2(\mathrm{AVG})$ is unbiased. The higher-order terms have a bias of order $(\Delta t)^4$. Figure 1 shows the absolute errors of the various v_n^2 estimators defined in Eqs. (35)–(38). The time step chosen is $\Delta t = 0.1$, $m = 1$, and $\omega = 1$. These parameters are in arbitrary units and correspond to approximately 20π integration points per period. We observe that $v_n^2(\mathrm{LF})$ oscillates around a negative value consistent with the prediction of Eq. (51), namely, -1.25×10^{-3}.

It is informative to calculate the effect of estimating the kinetic energy with $v_n^2(\mathrm{LF})$ for some typical harmonic oscillators in a MD simulation. Table I shows the results expected for the C–C, C–O, and H–N bonds as represented in the GROMOS 43A1 force field,[22] together with typical time steps. The relative kinetic energy bias is calculated as $\langle \Delta v_n^2(\mathrm{LF})\rangle / \langle v^2\rangle$, with the numerator evaluated using Eq. (51) and $\langle v^2\rangle$ the exact average squared velocity corresponding to an energy of $(1/2)k_BT$ at $T = 300$ K. In the case of a realistic motion, an amplitude prefactor has to be added to Eqs. (39)–(52), which are based on a trivial harmonic oscillator with position $x(t) = \cos(\omega t)$. For a total oscillator energy of k_BT, the amplitude is $x_0 = \sqrt{2k_BT/m\omega^2}$.

As a further example, we look at the interaction of two Lennard-Jones (LJ) particles. The parameters are chosen such that this example represents a typical interaction found in a MD simulation. The C6 and C12 parameters are those of the oxygen-oxygen interaction in a pair of water molecules, 2.617×10^{-3} and 2.634×10^{-6} kJ mol^{-1} nm,[12] respectively. Particles have a mass of 16 amu. The energy is set to k_BT at $T = 300$ K, and the time step is $\Delta t = 1$ fs, as often used in MD simulations. The initial positions and velocities are such that a centered collision is induced. No exact solution is available for the motion of LJ particles. We use instead the leapfrog algorithm to integrate the equations of motion and generate a trajectory. Expression $v_n^2(\mathrm{HO})$ [Eq. (37)] is evaluated *a posteriori* and serves as a reference for the other less accurate kinetic energy estimators. Figure 2 shows the behavior of the resulting errors for the kinetic energies calculated with the different v_n^2 estimators. The error of an estimator XX is calculated as $\Delta v_n^2(\mathrm{XX}) = v_n^2(\mathrm{XX}) - v_n^2(\mathrm{HO})$. It is again apparent that $v_n^2(\mathrm{HOnr})$ is the most accurate estimator. In addition, $v_n^2(\mathrm{LF})$ displays a negative error during the whole collision, indicating that this estimator is also biased on average for LJ interactions.

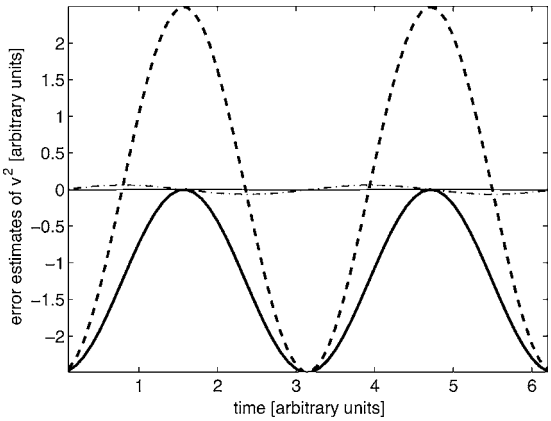

FIG. 1. Error of v^2 estimators for an harmonic oscillator with force constant $m\omega^2$, $\omega = 1$, $m = 1$, and $\Delta t = 0.1$. Thick line, $\Delta v_n^2(\mathrm{LF})$ [Eq. (35)]; dashed line, $\Delta v_n^2(\mathrm{AVG})$ [Eq. (36)]; dash-dotted line, $\Delta v_n^2(\mathrm{HOnr})$ [Eq. (38)]; and thin line, $\Delta v_n^2(\mathrm{HO})$ [Eq. (37)].

B. Realistic MD simulations

Based on the results of Table I, we can expect that the true kinetic energy of a molecular system modeled only by harmonic bonded terms could be approximately 3% above the value estimated with $v_n^2(\mathrm{LF})$ of Eq. (35). A realistic explicit solvent MD system, however, includes many other types of interactions, including softer angle and dihedral-angle potential energy terms, electrostatic as well as LJ interactions. Next, we assess the behavior of the different kinetic energy estimators for realistic MD systems. The instantaneous temperature of a dynamical system of N particles with N_{df} degrees of freedom is related to the average kinetic energy through

$$T(t) = \frac{1}{k_B N_{\mathrm{df}}} \sum_{i=1}^{3N} m_i v_i^2(t). \quad (53)$$

In systems with many degrees of freedom, the instantaneous errors of lower-order estimators for the squared velocities, such as $v_n^2(\mathrm{AVG})$ (36), tend to average out. Notwithstanding,

TABLE I. Relative kinetic energy bias due to the use of v_n^2(LF) [Eq. (35)] for typical bonds of the GROMOS 43A1 force field, (Ref. 22), with energy k_BT, $T=300$ K. Here, k_{bond} represents the harmonic constant and $m_\star$ the reduced mass of the two atoms.

Bond	k_{bond} (10^5 kJ/mol nm^2)	m_* (amu)	Period (fs)	Relative bias (%) $\Delta t=1$ fs	$\Delta t=0.5$ fs	$\Delta t=0.1$ fs
C–C	3.347	6.005	26.60	1.39	0.35	0.01
C–O	5.021	6.861	23.21	1.83	0.46	0.02
H–N	3.745	0.940	9.93	9.83	2.49	0.10

we have established that the estimator v_n^2(LF) (35) has a systematic bias, which remains apparent even when averaged over the system's degrees of freedom. In the following, we look at the impact on the temperature $T(t)$ resulting from the choice of the biased v_n^2(LF) versus nonbiased v_n^2(AVG) for estimating the full time step squared velocities.

The first system is a box of 1136 water molecules with one sodium ion. The bonds and the bond angle in the water molecules are constrained to their ideal values. There are no harmonic terms in this system, only electrostatic and LJ interactions. The second system is a large protein assembly of 605 residues, the A6 T-cell receptor bound to the HLA-A2 major histocompatibility complex (TCR-pMHC).[23] The protein is immersed in a rectangular periodic box with 25 981 water molecules, and the force field used is GROMOS 43A1.[22] All bonds involving hydrogens are constrained. In both cases, the simulation is done with the GROMACS 3.3 software,[24] which uses v_n^2(AVG) to couple the system temperature to a Nosé-Hoover thermostat[25] at 300 K and time constant of 0.15 ps. The electrostatic interactions are treated with the particle-mesh Ewald scheme[26] with a direct-space cutoff of 0.8 nm. The LJ interactions are updated every step up to a cutoff distance of 0.8 nm, and every five steps up to a cutoff distance of 1.4 nm. The integration time step is 1 fs, and data are averaged over 100 ps. Table II shows the average temperature calculated using v_n^2(LF) (35). As expected from the sign of expression (32) it lies below the true temperature. This effect is more pronounced for the protein system, which contains harmonic bonds.

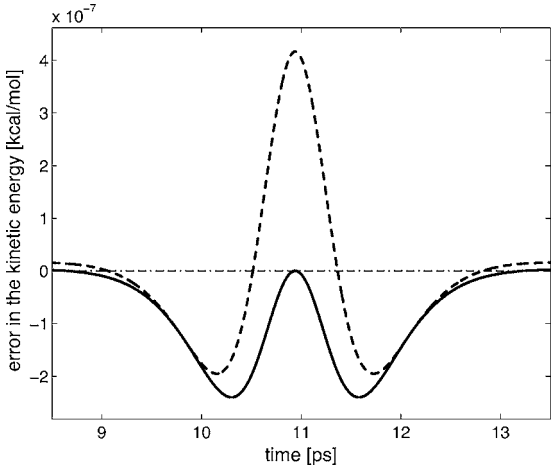

FIG. 2. Error of kinetic energy estimators for a collision of two Lennard-Jones particles, water oxygens with energy k_BT for $T=300$ K, with $\Delta t=1$ fs. Thick line, $(1/2)m\Delta v_n^2$(LF) [Eq. (35)]; dashed line, $(1/2)m\Delta v_n^2$(AVG) [Eq. (36)]; and dashed-dotted line, $(1/2)m\Delta v_n^2$(HOnr) [Eq. (38)]. Here, $(1/2)mv_n^2$(HO) [Eq. (37)] is the reference.

V. DISCUSSION

We have suggested the use of the more accurate expression v_n^2(HOnr) (38) or of the unbiased one v_n^2(AVG) (36) instead of the standard leapfrog one v_n^2(LF) (35) in MD simulation. This will improve the accuracy of molecular or system properties that depend on the squared velocities at full time steps. Often, the accuracy of an integration scheme is evaluated by considering the degree of conservation of the total energy, e.g., $x^2(t)+v^2(t)$ for the one-dimensional harmonic oscillator. Using a low-order or biased formula for estimating the squared velocities might lead to erroneous conclusions[27] regarding the accuracy of the formula for propagating the positions.[28]

The leapfrog integration scheme can also be formulated for stochastic dynamics (SD) simulation[29] in which instead of Newton's equation (5), the Langevin equation,

$$\dot{v}_i(t) = \frac{1}{m_i} f_i(t) + \frac{1}{m_i} f_i^{\text{stoch}}(t) - \gamma_i v_i(t), \tag{54}$$

is integrated forward in time. The particle i is subjected to a force f_i^{stoch} and a frictional force, the size of which is determined by the particle friction coefficient γ_i. Because of the velocity term occurring in Eq. (54), the invariance of the equation of motion under time reversal is lost and the SD leapfrog algorithm becomes more complex than the MD one.[22,30] Yet, the use of v_n^2(HOnr) (38) or v_n^2(AVG) (36) instead of v_n^2(LF) (35) is also recommended in this case. We note that a formal discussion of the precision of different expressions for the squared velocity is presented in Ref. 31, however, only for the case of harmonic forces in either the low friction or the high friction limit but not for general forces and arbitrary collision frequencies.

In order to speed up MD or SD simulations in which motions of different, well separated time scales occur, the

TABLE II. Average temperature calculated with the biased expression v_n^2(LF) [Eq. (35)] from MD simulations at 300 K, with $\Delta t=1$ fs, and using an unbiased thermostat feedback.

System	Number of atoms Solute	Solvent	Feedback temperature (K)	LF temperature (K)
Na^+ in water	1	3408	300.00	299.54
TCR-pMHC	6286	77 943	300.00	299.35

TABLE III. Unbiased temperature and pressure from MD simulations with reference temperature 300 K and reference pressure 1 atm but using the biased v_n^2(LF) [Eq. (35)] for the thermostat and barostat feedback.

System	Δt (fs)	True temperature (K)	Feedback pressure (bar)	True pressure (bar)
Na^+ in water	1	300.46	1.01	5.14
Na^+ in water	2	301.83	1.24	17.83
TCR-pMHC	1	300.65	1.05	6.97
protein		302.23		
solvent		300.48		

use of multiple–time-step algorithms has been proposed.[19,32,33] The use of higher-accuracy expressions for the squared velocities should also be considered in those cases.

As shown in Table II, using the biased expression v_n^2(LF) (35) leads to a systematic underevaluation of the temperature of the system (at full time steps). This effect impacts the simulation if the biased full time step temperature is used to couple the system to a heat bath. In such a constant temperature simulation, a thermostat is used to keep the average biased temperature close to the reference temperature (300 K). Table III illustrates that in this case, the true average temperature of the system systematically lies above the reference temperature. The difference is more important in the protein itself, where harmonic bonds are present.

The consequences are more severe if the system is in addition subjected to pressure coupling. In this case, the volume of the simulation box is scaled in such a way that the average internal pressure of the system matches a reference pressure. As a consequence of the virial theorem, the internal scalar pressure at time t, which can be used for pressure coupling in isotropic systems, is calculated as

$$p(t) = \frac{1}{3V(t)} \sum_{i=1}^{3N} [m_i v_i^2(t) + x_i(t) f_i(t)]. \tag{55}$$

Here, the squared velocities should be taken at full time steps since the positions and forces come at full time steps. The second term usually being negative for condensed phase systems, a small absolute error on $v_i^2(t)$ may result in a relatively higher error on $p(t)$. Table III shows the true average pressure in molecular simulations in which the biased v_n^2(LF) is fed to the barostat. The same Nosé-Hoover thermostat as before was used, together with a Parrinello-Rahman[34,35] barostat with time constant of 0.5 ps.

VI. CONCLUSION

We have derived a higher-order expression for full time step squared velocities in the leapfrog algorithm, where only velocities at half time steps are accurately known. This expression has one term of zeroth order in Δt and two first-order terms. This decomposition allows us to show that the usual leapfrog expression for the squared velocities is a biased estimator. Indeed it is equivalent to the first and second terms only, and the latter is shown to be always negative. An appropriate practical expression for the full time step squared velocity is either the first, zero-order term only or the expression including both first-order terms. The zero-order term is readily available as average of the preceding and following squared half step velocities. The full first-order expression can only be evaluated at the cost of an approximation which destroys its formal time-reversal invariance.

We have demonstrated with one-dimensional harmonic and Lennard-Jones systems that the higher-order expression is more accurate. While the zero-order term shows larger instantaneous errors, it is unbiased. In practical molecular systems with many degrees of freedom, the instantaneous errors cancel out, but the bias of the standard leapfrog estimator remains. We assessed the impact of using this biased estimator (as it is done in some state-of-the-art MD software packages) on temperature and pressure regulation of molecular systems. While the induced temperature error is within 1%, the resulting pressure is manyfold higher than the reference pressure, by a factor of between 5 and 7 with a time step of 1 fs and a factor of 14 with a time step of 2 fs. For systems with numerous degrees of freedom, which allow for error compensation, the use of the zero-order unbiased formula (36) for the full time step squared velocities seems appropriate. On the other hand, for low-dimensional systems, with less error compensation, the use of the higher-order formula (38) is recommended.

ACKNOWLEDGMENTS

The authors thank Daniel Trzesniak and Philippe H. Hünenberger for stimulating discussions regarding the accuracy of MD algorithms. One of the authors (M.A.C.) acknowledges the support of U. Röthlisberger and O. Michielin.

APPENDIX: LEAPFROG INTEGRATOR FOR COUPLED SYSTEMS

In this appendix, we show that the issues related to the estimation of v_n^2 can be bypassed for the coupling of the system to a heat bath. If appropriate time-reversible integration schemes are chosen, only the temperature at half time steps is needed for the thermostat feedback. As a first example, we consider the weak coupling scheme of Berendsen *et al.*,[36]

$$v_{n+1/2} = v_{n-1/2} + \frac{1}{m} f_n \Delta t, \tag{A1}$$

$$v_{n+1/2}^{\text{scaled}} = v_{n+1/2} \sqrt{1 + \frac{\Delta t}{\tau_T} \left[\frac{T_0}{T_{n+1/2}} - 1 \right]}, \tag{A2}$$

$$x_{n+1} = x_n + v_{n+1/2}^{\text{scaled}} \Delta t. \tag{A3}$$

The parameter τ_T sets the time scale of the first-order velocity relaxation induced by the scaling term, and T_0 is the reference temperature. We see that the temperature $T_{n+1/2}$, calculated using Eq. (53) with the velocities $v_{n+1/2}$, is used.

As a further example, we consider the Nosé-Hoover[25] (NH) thermostated dynamics,

$$\dot{r}_i(t) = v_i(t), \tag{A4}$$

$$\dot{v}_i(t) = \frac{f_i(t)}{m_i} - \xi v_i, \tag{A5}$$

$$\dot{\xi}(t) = \frac{1}{Q}(T(t) - T_0). \tag{A6}$$

Here, Q is a pseudomass regulating the time scale of the second-order relaxation induced by the coupling variable ξ. A consistent leapfrog scheme can be obtained[37–39] for the NH dynamics using the same approach as for the standard leapfrog [Eqs. (20) and (21)]. This approach involves expanding each of the above variables in Taylor series in $\Delta t/2$ and $-\Delta t/2$ at time point t, subtracting the resulting expansions, and rearranging. One finds

$$\xi_n = \xi_{n-1} + \frac{1}{Q}[T_{n-1/2} - T_0]\Delta t, \tag{A7}$$

$$v_{n+1/2} = \frac{v_{n-1/2}[1 - \xi_n \Delta t/2] + (1/m) f_n \Delta t}{1 + \xi_n \Delta t/2}, \tag{A8}$$

$$x_{n+1} = x_n + v_{n+1/2}\Delta t. \tag{A9}$$

It is clear that this integrator is fully time reversible. Again, only $T_{n-1/2}$ is needed, which is accurately calculated from the velocities at half time steps. Only fully consistent integrators such as Eqs. (A7)–(A9) should be used in MD software. Alternatively, one could choose the integrators for extended dynamics proposed by Martyna *et al.*,[40] based on the Trotter decomposition of the Liouville operator. Note that these integrators are similar to the velocity Verlet[41] and thus provide accurate velocities at full time steps.

[1] H. Goldstein, *Classical Mechanics* (Addison Wesley, Reading, MA, 1950).
[2] C. W. Gear, *Numerical Initial Value Problems in Ordinary Differential Equations* (Prentice-Hall, Engelwood Cliffs, NJ, 1971).
[3] L. Lapidus and J. H. Seinfeld, *Numerical Solutions of Ordinary Differential Equations* (Academic, New York, 1971).
[4] G. Dahlquist and A. Björk, *Numerical Methods* (Prentice-Hall, Engelwood Cliffs, NJ, 1974).
[5] H. J. C. Berendsen and W. F. van Gunsteren, *Proceedings of the International School of Physics "Enrico Fermi," Course 97*, edited by G. Ciccotti and W. Hoover (North-Holland, Amsterdam, 1986), p. 43.
[6] W. F. van Gunsteren and H. J. C. Berendsen, Mol. Phys. **34**, 1311 (1977).
[7] J. Delambre, Mem. Acad. Turin **5**, 143 (1790).
[8] C. Störmer, Arch. Sci. Phys. Nat. **24**, 5113221 (1907).
[9] L. Verlet, Phys. Rev. **159**, 98 (1967).
[10] D. Levesque and L. Verlet, J. Stat. Phys. **72**, 519 (1993).
[11] R. W. Hockney, Methods Comput. Phys. **9**, 136 (1970).
[12] P. J. Channell and C. Scovel, Nonlinearity **3**, 231 (1990).
[13] J. M. Sanz-Serna, Acta Numerica **1**, 243 (1991).
[14] H. Yoshida, Celest. Mech. Dyn. Astron. **56**, 27 (1993).
[15] B. Leimkuhler, *Simulating Hamiltonian Dynamics* (Cambridge University Press, Cambridge, 2005).
[16] S. Toxvaerd, Phys. Rev. E **50**, 2271 (1994).
[17] R. D. Skeel, G. Zhang, and T. Schlick, SIAM J. Sci. Comput. (USA) **18**, 203 (1997).
[18] S. A. Chin and S. R. Scuro, Phys. Lett. A **342**, 397 (2005).
[19] M. Tuckerman, B. J. Berne, and G. J. Martyna, J. Chem. Phys. **97**, 1990 (1992).
[20] S. Toxvaerd, J. Chem. Phys. **99**, 2277 (1993).
[21] M. Tuckerman, B. J. Berne, and G. J. Martyna, J. Chem. Phys. **99**, 2278 (1993).
[22] W. F. van Gunsteren, S. R. Billeter, A. A. Eising, P. H. Hünenberger, P. Krüger, A. E. Mark, W. R. P. Scott, and I. G. Tironi, *Biomolecular Simulation: The GROMOS96 Manual and User Guide* (Vdf Hochschulverlag AG an der ETH Zürich, Zürich, Switzerland, 1996).
[23] D. N. Garboczi, P. Ghosh, U. Utz, Q. R. Fan, W. E. Biddison, and D. C. Wiley, Nature (London) **384**, 134 (1996).
[24] D. van der Spoel, E. Lindahl, B. Hess, G. Groenhof, A. E. Mark, and H. J. C. Berendsen, J. Comput. Chem. **26**, 1701 (2005).
[25] W. G. Hoover, Phys. Rev. A **31**, 1695 (1985).
[26] U. Essmann, L. Perera, M. L. Berkowitz, T. Darden, H. Lee, and L. G. Pedersen, J. Chem. Phys. **103**, 8577 (1995).
[27] M. Levitt and H. Meirovitch, J. Mol. Biol. **168**, 617 (1983).
[28] J. Åqvist, W. F. van Gunsteren, M. Leijonmarck, and O. Tapia, J. Mol. Biol. **183**, 461 (1985).
[29] W. F. van Gunsteren and H. J. C. Berendsen, Mol. Simul. **1**, 173 (1988).
[30] W. R. P. Scott, P. H. Hünenberger, I. G. Tironi, A. E. Mark, S. R. Billeter, J. Fennen, A. E. Torda, T. Huber, P. Krüger, and W. F. van Gunsteren, J. Phys. Chem. A **103**, 3596 (1999).
[31] R. W. Pastor, B. R. Brooks, and A. Szabo, Mol. Phys. **65**, 1409 (1988).
[32] W. F. van Gunsteren and H. J. C. Berendsen, Angew. Chem., Int. Ed. Engl. **29**, 992 (1990).
[33] X. Qian and T. Schlick, J. Chem. Phys. **116**, 5971 (2002).
[34] M. Parrinello and A. Rahman, J. Appl. Phys. **52**, 7182 (1981).
[35] S. Nosé and M. L. Klein, Mol. Phys. **50**, 1055 (1983).
[36] H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, A. DiNola, and J. R. Haak, J. Chem. Phys. **81**, 3684 (1984).
[37] S. Toxvaerd, Mol. Phys. **72**, 159 (1991).
[38] S. Toxvaerd, Phys. Rev. E **47**, 343 (1993).
[39] B. L. Holian, A. F. Voter, and R. Ravelo, Phys. Rev. E **52**, 2338 (1995).
[40] G. J. Martyna, M. E. Tuckerman, D. J. Tobias, and M. L. Klein, Mol. Phys. **87**, 1117 (1996).
[41] W. C. Swope, H. C. Andersen, P. H. Berens, and K. R. Wilson, J. Chem. Phys. **76**, 637 (1982).