Langevin stabilization of molecular-dynamics simulations of polymers by means of quasisymplectic algorithms

L. Larini
Dipartimento di Fisica "Enrico Fermi," Università di Pisa, Largo B. Pontecorvo 3, I-56127 Pisa, Italy and INFM-CRS SOFT, Largo B. Pontecorvo 3, I-56127 Pisa, Italy

R. Mannella
Dipartimento di Fisica "Enrico Fermi," Università di Pisa, Largo B. Pontecorvo 3, I-56127 Pisa, Italy and INFM, UdR Pisa, Largo B. Pontecorvo 3, I-56127 Pisa, Italy

D. Leporini[a)]
Dipartimento di Fisica "Enrico Fermi," Università di Pisa, Largo B. Pontecorvo 3, I-56127 Pisa, Italy and INFM-CRS SOFT, Largo B. Pontecorvo 3, I-56127 Pisa, Italy

(Received 1 August 2006; accepted 9 January 2007; published online 8 March 2007)

Algorithms for the numerical integration of Langevin equations are compared in detail from the point of view of their accuracy, numerical efficiency, and stability to assess them as potential candidates for molecular-dynamics simulations of polymeric systems. Some algorithms are symplectic in the deterministic frictionless limit and prove to stabilize long time-step integrators. They are tested against other popular algorithms. The optimal algorithm depends on the main goal: accuracy or efficiency. The former depends on the observable of interest. A recently developed quasisymplectic algorithm with great accuracy in the position evaluation exhibits better overall accuracy and stability than the other ones. On the other hand, the well-known BrünGer-Brooks-Karplus [Chem. Phys. Lett. **105**, 495 (1982)] algorithm is found to be faster with limited accuracy loss but less stable. It is also found that using higher-order algorithms does not necessarily improve the accuracy. Moreover, they usually require more force evaluations per single step, thus leading to poorer performances. © *2007 American Institute of Physics*.
[DOI: 10.1063/1.2464095]

I. INTRODUCTION

The physics of macromolecules is a major research field with extremely important applications to both the material science and the biology. In recent years, computer simulations have developed into a powerful tool for studying the dynamic, structural, and topological features of macromolecular systems. A major challenge in these studies is the wide range of length and time scales characterizing the molecular structure and the dynamics, respectively. From this respect, one is interested in algorithms which may make use of long time steps in the integration of the equations of motion, in order to investigate the slow dynamics, with limited or no loss of accuracy to sample the short time dynamics in an accurate way. That issue will be discussed in the present paper by referring us to the case of polymers, where a large number of monomers are linked together by covalent bonds to form linear chains or more complex architectures.[1–4]

Heretofore, molecular-dynamics (MD) simulations of dense polymer systems have been limited to relatively long chains (usually not exceeding 1000 monomers for coarse-grained models and even less for atomistic models) and time spans rarely exceeding a few hundred of nanoseconds.[5] The Monte Carlo (MC) method offers an interesting alternative to MD. Through the design of clever moves, configurational sampling can be dramatically enhanced. MC moves such as concerted rotation,[6] configurational bias,[7,8] and internal configurational bias[9] have thus successfully addressed the problem of equilibrating polymer systems with moderate chain lengths, however, even these moves were proven incapable of providing equilibration when applied to polymer melts of molecular length exceeding about 100 monomers. A solution to this problem was given by the development and efficient implementation of a chain connectivity altering MC move, i.e., end bridging (EB).[10–12] Using EB, atomistic systems consisting of a large number of long chains, up to 1000 monomers, have been simulated in full atomistic detail.[11,12] Similar efforts, employing chain connectivity altering segmental rearrangements, include the cooperative motion algorithm[13] used in lattice-based simulations of complex polymer systems and an off-lattice MC study of the interphase between crystals with freely rotating chains.[14] Despite its remarkable efficiency in equilibrating long chain polymer melts, EB suffers from three shortcomings. (a) It cannot equilibrate monodisperse polymer melts; a finite degree of polydispersity is necessary for the move to function. While this is not a drawback in modeling industrial polymers, which are typically polydisperse, the ability to equilibrate strictly the monodisperse polymers is highly desirable for comparing simulations with the theory and the experiments on anionically synthesized model systems. (b) It relies on the presence of chain ends. Thus, it does not offer itself for dense

[a)]Electronic mail: dino.leporini@df.unipi.it

phases of chains with nonlinear architectures containing long strands between branch points such as H-shaped polymers, for cyclic molecules, or for model polymer systems with infinite chain length. (c) Its performance drops considerably as the stiffness of chains increases or in the presence of chain orientation. In order to overcome these problems, new connectivity altering moves, involving two bridging trimers among four properly chosen monomers along one or two chains in the system, were developed.[15,16]

Polymer physics is inherently multiscale in nature, i.e., microscopic interactions are strongly coupled to meso- and macroscopic properties.[1–4] As noted above, despite the increasing computational power and ongoing efforts to enhance the efficiency of MD integration algorithms,[17–21] all-atom MD simulations are often incapable to cover the time scales of dense polymer systems, especially for long chains.[5] However, in some cases the need of the huge amount of detailed information being provided by that approaches may be questioned (e.g., the chemical details affect the prefactors of the universal power laws only). This motivates the strategy to perform a systematic coarse graining of the polymer chain to retain the relevant degrees of freedom only. Recently, great efforts have been devoted to the development of multiscale approaches, where different parts of the system are modeled with different accuracies.[22–27] Similar procedures are well known in polymer physics since long time, e.g., in the Rouse model dealing with short chains where, given a particular chain, the effect of the surroundings is accounted for by a stochastic background and the chain dynamics is described by suitable Langevin equations.[2,3] The Rouse model describes the long time behavior very well and belongs to the class of models where the stochastic behavior pictures the coupling with a thermal bath (alternatively, one may modify the equations of motion by coupling the system to an additional degree of freedom[30–33]). In the Langevin dynamics formalism, the degrees of freedom of the bath are eliminated by using the Mori-Zwanzig projection technique.[34,35] The result is a set of stochastic differential equations describing the dynamic state of the target system.[36] Langevin dynamics has been also used to accelerate the exploration of the high-dimensional configuration space of macromolecules.[37]

Within an approach based on stochastic differential equations, symplectic algorithms are noteworthy. Symplectic integrators are numerical integration schemes for Hamiltonian systems, which conserve the symplectic two-form $dp \wedge dq$ exactly, so that $(q(0),p(0)) \rightarrow (q(\tau),p(\tau))$ is a canonical transformation. For both explicit and implicit integrators, it was shown that the discrete mapping obtained describes the exact time evolution of a slightly perturbed Hamiltonian system and thus possesses the perturbed Hamiltonian as a conserved quantity. That feature is of interest in common integration schemes which conserve not the complete Hamiltonian but some other quantity which slightly differs from it.[38–42] Symplectic algorithms guarantee that, in spite of the local truncation error, the total energy (which should be conserved exactly in the original flow) exhibits limited errors growing in time, the so-called secular errors. If the integrator is not symplectic, secular errors of the total energy are usually observed. Dedicated symplectic algorithms were derived for studying the monodimensional motion of a single Brownian particle.[38,43] In that case the equation of motion is

$$\dot{v} = F(x) - \gamma v + \xi(t), \tag{1}$$

where the mass of the particle is unitary, x is the position, $v=\dot{x}$ the velocity, $F(x)$ the force, γ is the damping factor, and $\xi(t)$ is a random Gaussian noise, with zero average and delta correlations,

$$\langle \xi(t)\xi(s) \rangle = 2\gamma T \delta(t-s), \tag{2}$$

where T is the temperature of the system. It will be proven that the above dedicated algorithms are also well suited for multiparticle systems in the usual three-dimensional space. To this aim, the dynamics of a single polymer chain in dilute solution will be considered.[3,4,44] Trajectories were generated by the above symplectic algorithms and compared with some popular alternatives.

The paper is organized as follows. In the Sec. II the symplectic algorithms of interest are presented together with the alternative ones. In Sec. III the polymer model and the details of the simulations are given. The results are discussed in Sec. IV. Section V summarizes the conclusions.

II. ALGORITHMS

First, let us discuss symplectic integrators and consider the following problem.[40,45] Let A and B be noncommutative operators and τ be a small real number. For a given n integer (later, to be identified with the order of the integrator), find a set of real numbers $(c_1,c_2,\ldots,c_k)$ and $(d_1,d_2,\ldots,d_k)$ such as

$$\exp[\tau(A+B)] = \prod_{i=1}^{k} \exp(c_i \tau A)\exp(d_i \tau B) + o(\tau^{n+1}). \tag{3}$$

As an example, for $n=2$ and $k=2$, one finds

$$\exp[\tau(A+B)] = \exp\left(\tfrac{1}{2}\tau A\right)\exp(\tau B)\exp\left(\tfrac{1}{2}\tau A\right) + o(\tau^3). \tag{4}$$

The above general problem is directly related to the symplectic integrator of a Hamiltonian $H(p,q)=T(p)+V(q)$.[40]

Let us consider the Liouville operator L for a system with f degrees of freedom,

$$iL = \{\ldots,H\} = \sum_{j=1}^{f} \left[\dot{x}_j \frac{\partial}{\partial x_j} + F_j \frac{\partial}{\partial p_j} \right], \tag{5}$$

where x_j and p_j are the positions and conjugate momenta of the system, respectively, F_j is the force on the jth degree of freedom, and $\{\ldots,\ldots\}$ is the Poisson bracket of the system. L is a linear Hermitian operator on the space of square integrable functions of Γ, with $\Gamma=\{x_j,p_j\}$. The state of the system at time t is given by

$$\Gamma(t) = U(t)\Gamma(0), \tag{6}$$

where the classical propagator is defined as

$$U(t) = \exp(iLt). \tag{7}$$

In a symplectic algorithm one key step is the separation of the Liouvillian in two terms L_1 and L_2 such as $L=L_1+L_2$. For

second-order integrators ($n=2$) the Trotter theorem[46] ensures that[47]

$$\exp[i(L_1+L_2)t] = \{\exp[iL_1(h/2)]\exp[iL_2(h)]\exp[iL_1(h/2)]\}^P, \quad (8)$$

where $h=t/P$. From this one defines the discrete time propagator as

$$G(h) = U_1\left(\frac{h}{2}\right)U_2(h)U_1\left(\frac{h}{2}\right) = \exp[iL_1(h/2)]\exp[iL_2(h)]\exp[iL_1(h/2)]. \quad (9)$$

Notice that the propagator $G(h)$ is time reversible. As an example, the Liouvillian, Eq. (5), may be separated as[48]

$$iL_1 = \sum_{j=1}^{f} F_j \frac{\partial}{\partial p_j}, \quad iL_2 = \sum_{j=1}^{f} \dot{x}_j \frac{\partial}{\partial x_j}. \quad (10)$$

The above choice leads to the velocity Verlet (VV) algorithm, which in one dimension takes the form[49]

$$x(t+h) = x(t) + hv(t) + \tfrac{1}{2}h^2\dot{v}(t),$$
$$v(t+h) = v(t) + \tfrac{1}{2}h[\dot{v}(t) + \dot{v}(t+h)], \quad (11)$$

where h is the integration time step and $\dot{v}$ is given by the deterministic equation of motion. If iL_1 and iL_2 are exchanged, the position Verlet (PV) algorithm is obtained.[47] Both algorithms are correct to second order.

The deterministic VV algorithm, Eq. (11), may be augmented to the corresponding stochastic VV (SVV) algorithm as (using the prescription in Ref. 52)

$$x(t+h) = x(t) + hv(t) + \tfrac{1}{2}h^2a(t),$$
$$v(t+h) = v(t) + \tfrac{1}{2}h[a(t) + a(t+h)], \quad (12)$$

where $a(t+h)$ is the acceleration defined as

$$v\left(t+\frac{h}{2}\right) = v(t) + \frac{1}{2}ha(t),$$
$$a(t+h) = F(x(t+h)) - \gamma v\left(t+\frac{h}{2}\right) + \xi, \quad (13)$$

where ξ is a random Gaussian variable with $\sqrt{2T\gamma h}$ standard deviation and zero average.

To summarize, the basic ingredients of a *deterministic* symplectic algorithm are the optimum choice of the separation $L=L_1+L_2$ and the subsequent explicit form of Eq. (3) to get the numerical integration scheme with the aid of Eqs. (6) and (7).

For a given deterministic symplectic scheme with n order, a corresponding quasisymplectic *stochastic* algorithm may be found.[39] As an example, let us consider the deterministic leap-frog algorithm,[38,39]

$$\bar{x} = x(t) + \frac{h}{2}v(t),$$
$$v(t+h) = v(t) + hF(\bar{x}),$$
$$x(t+h) = \bar{x} + \frac{h}{2}v(t+h). \quad (14)$$

The corresponding stochastic algorithm, the so-called symplectic low order (SLO) algorithm, is described by the equations[38]

$$\tilde{x} = x(t) + \frac{h}{2}v(t),$$
$$v(t+h) = c_2[c_1v(t) + hF(\tilde{x}) + d_1\xi], \quad (15)$$
$$x(t+h) = \tilde{x} + \frac{h}{2}v(t+h),$$

where ξ is a random Gaussian variable with unit standard deviation and zero average and

$$c_1 = 1 - \frac{\gamma h}{2}, \quad c_2 = \frac{1}{1+\gamma h/2}, \quad d_1 = \sqrt{2T\gamma h}.$$

The SLO algorithm is able to reproduce the probability distribution of position up to an order $O(h)$ (not to be confused with the order of the integrator). It can be further improved by deriving an algorithm with order $O(h^2)$. The details are given elsewhere.[38] The integration scheme, the so-called symplectic high order (SHO) algorithm, is as follows:

$$x_1 = x(t) + \frac{h}{4}v(t),$$
$$v_1 = c_2\left[c_1v(t) + \frac{h}{2}F(x_1) + \sqrt{\gamma Th}(a_1\xi_1 + a_2\xi_2)\right],$$
$$x_2 = x_1 + \frac{h}{2}v_1, \quad (16)$$
$$v(t+h) = c_2\left[c_1v_1 + \frac{h}{2}F(x_2) + \sqrt{\gamma Th}(b_1\xi_1 + b_2\xi_2)\right],$$
$$x(t+h) = x_2 + \frac{h}{4}v(t+h),$$

with

$$c_1 = 1 - \frac{\gamma h}{4},$$
$$c_2 = \frac{1}{1+\gamma h/4},$$
$$a_1 = -1.069\,186\,004\,330\,706\,5\ldots,$$
$$a_2 = -0.153\,323\,040\,701\,989\,3\ldots,$$
$$b_1 = 0.304\,491\,312\,885\,406\,5\ldots,$$
$$b_2 = -1.036\,316\,412\,609\,579\,0\ldots.$$

Here ξ_1 and ξ_2 are two uncorrelated random Gaussian variables with unit standard deviation and zero average. Note

that although this algorithm uses two random variables, the weight they have in the scheme has been crafted to yield the same amount of coupling with the bath.

One is also interested in testing a fourth-order algorithm, which is a derivation of a fourth-order Hamiltonian Runge-Kutta (HRK4) scheme,[39,50]

$$v_i = c_i^{(2)}[c_i^{(1)} v_{i-1} + b_i(F(x_{i-1}) + d_1\xi)],$$
$$x_i = x_{i-1} + h a_i v_i, \tag{17}$$

where $1 \leqslant i \leqslant 4$ and

$$x_0 = x(t), \quad v_0 = v(t),$$
$$x_4 = x(t+h), \quad v_4 = v(t+h),$$
$$a_1 = 0.515\,352\,837\,431\,123,$$
$$b_1 = 0.134\,496\,199\,277\,431,$$
$$a_2 = -0.085\,782\,019\,412\,974,$$
$$b_2 = -0.224\,819\,803\,079\,421,$$
$$a_3 = 0.441\,583\,023\,616\,467,$$
$$b_3 = 0.756\,320\,000\,515\,668,$$
$$a_4 = 0.128\,846\,158\,365\,384,$$
$$b_4 = 0.334\,003\,603\,286\,321,$$
$$c_i^{(1)} = 1 - \frac{\gamma h}{2} b_i,$$
$$c_i^{(2)} = \frac{1}{1 + \frac{\gamma h}{2} b_i}.$$

In addition to the SVV algorithm, other two additional benchmarks will be also considered. First, the popular Brünger-Brooks-Karplus (BBK) algorithm,[51]

$$\tilde{v} = v(t) + \frac{h}{2} F(x(t)),$$
$$x(t+h) = x(t) + \frac{[1 - \exp(-\gamma h)]}{\gamma} \tilde{v} + \sqrt{2T/\gamma}\,\zeta_2, \tag{18}$$
$$v(t+h) = \exp(-\gamma h)\tilde{v} + \frac{h}{2} F(x(t+h)) + \sqrt{2T\gamma}\,\zeta_1.$$

The stochastic variables ζ_1 and ζ_2 are defined as

$$\zeta_1 = \sqrt{\tau_2}\,\xi_1,$$
$$\zeta_2 = \frac{\tau_1 - \tau_2}{\sqrt{\tau_2}} \xi_1 + \sqrt{h - \frac{\tau_1^2}{\tau_2}}\,\xi_2,$$

where ξ_1 and ξ_2 are two uncorrelated Gaussian variables with zero average and unit standard deviation and

$$\tau_1 = \frac{1}{\gamma}[1 - e^{-\gamma h}],$$
$$\tau_2 = \frac{1}{2\gamma}[1 - e^{-2\gamma h}].$$

Finally, one will also consider the well-known algorithm (to be referred to as Li, from "liquids"),[52]

$$x(t+h) = x(t) + c_1 h v(t) + c_2 h^2 F(x(t)) + \xi_1,$$
$$v(t+h) = c_0 v(t) + (c_1 - c_2) h F(x(t)) + c_2 F(x(t+h)) + \xi_2, \tag{19}$$

where

$$c_0 = e^{-\gamma h}, \quad c_1 = \frac{1 - c_0}{\gamma h}, \quad c_2 = \frac{1 - c_1}{\gamma h}.$$

The above algorithm was analyzed in depth elsewhere.[38]

III. MODEL AND DETAILS OF SIMULATION

The system to be used to test the different algorithms listed in Sec. II is a model of a single polyethylene chain with $M = 20$ monomers in solution. It was studied in detail by the present[53–55] and other authors.[56–58] The chain is described as a sequence of beads, where each bead pictures a single methylene CH_2 group. No distinction is made between internal methylene CH_2 groups and terminal methyl CH_3 groups in order to obtain a slight improvement in efficiency.

The local interactions shaping the chain are defined by the potentials,

$$U_{\text{bond}}(r) = k_r (r - r_0)^2, \tag{20}$$

$$U_{\text{angle}}(\theta) = k_\theta (\cos\theta - \cos\theta_0)^2, \tag{21}$$

$$U_{\text{torsion}}(\phi) = k_1(1 - \cos\phi) + k_2(1 - \cos 2\phi) + k_3(1 - \cos 3\phi). \tag{22}$$

$U_{\text{bond}}(r)$ is a harmonic spring potential defined for every couple of adjacent beads, r being their distance, and r_0 the equilibrium bond length. $U_{\text{angle}}(\theta)$ is defined for every triplet of adjacent beads, θ being the angle between the corresponding bonds and θ_0 its equilibrium value. $U_{\text{torsion}}(\phi)$ is defined for every quadruplet of adjacent beads and ϕ is the dihedral angle between the planes defined by the corresponding three adjacent bonds. Finally, couples of beads which are not interacting by any of the preceding potentials will interact via a Lennard-Jones potential

$$U_{\text{LJ}}(r) = 4\epsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right] + U_{\text{cut}}. \tag{23}$$

The parameter U_{cut} shifts the potential to zero at the cutoff distance $r_{\text{cut}} = 2^{1/6}\sigma$ (that is, the Lennard-Jones miminum). $U_{\text{LJ}} = 0$ for $r \geqslant r_{\text{cut}}$. The values of the various parameters are

TABLE I. Model parameters.

Parameter	Value	
	Reduced units	SI units
ϵ	1	0.112 kcal/mol
σ	1	4.04 Å
m	1	14.03 g/mol
γ	1	0.455 s^{-1}
k_r	51 005	350 kcal/mol $Å^2$
r_0	0.38	1.53 Å
k_θ	535.71	60 kcal/mol
θ_0	109°	109°
k_1	7.1429	0.8 kcal/mol
k_2	−3.8393	−0.43 kcal/mol
k_3	14.4643	1.62 kcal/mol

listed in Table I. The force field ensures that the bond length and the bond angle are virtually constant. Reduced units will be used: σ is the unit length, ϵ is the unit energy, and the united-atom mass m is the unit mass. The corresponding time and temperature units are given by $t^*=2.21$ ps and $T^*=56.3$ K. The motion is described by the Langevin equation,[28,29]

$$\ddot{\mathbf{r}}_i = -\nabla_i U - \gamma \dot{\mathbf{r}}_i - \boldsymbol{\xi}_i, \tag{24}$$

where $\mathbf{r}_i$ denotes the position vector of the ith bead, $\nabla_i U$ is the sum the internal forces acting on it, $-\gamma\dot{\mathbf{r}}_i$ is the frictional force, and $\boldsymbol{\xi}_i$ is a Gaussian noise with zero average and delta like correlation;

$$\langle \boldsymbol{\xi}_i(t) \cdot \boldsymbol{\xi}_j(t') \rangle = 6\gamma k_B T \delta_{ij} \delta(t-t'). \tag{25}$$

It is understood that the different Cartesian components are mutually independent for $i=j$. All the simulations have been performed according to the following protocol. The system was initially equilibrated at the temperature $T=9$ in MD units. Then, a single configuration was singled out and used as seed for all the production runs with the different algorithms.

About the choice of the model, described by Eqs. (20)–(25), a point warrant emphasis. This model permits us to check the performance of the algorithm on a system composed of different potentials mutually interacting. With respect to a harmonic solvable model, it presents the advantage that anharmonic features can also be addressed. Another aspect, more technical, is related to the equipartition of the energy, which is reached very slowly in the harmonic approximation. In this situation a well equilibrated configuration can alleviate that drawback.[36]

IV. RESULTS AND DISCUSSION

Below, we discuss our results splitting them into two parts pointing out the global and detailed aspects of the algorithms.

A. Global aspects

1. Stability and numerical efficiency

To compare the different algorithms, the integration time step h was changed to identify the largest value $h_{\max}$ above which each algorithm is unstable. The algorithm is defined as unstable if it crashes before 10^4 iterations. Usually this happens when the algorithm is unable to integrate in a correct way the faster degrees of motion (i.e., the bond potential in the present case); in that case a fast increase in global energy (driven by the increase in the bond energy) is observed which leads the simulations to crash. From this respect, different behaviors were observed. SVV, SLO, BBK, and Li become unstable for $h_{\max} \cong 3.4\times10^{-3}$, whereas SHO ($h_{\max}^{\mathrm{SHO}} \cong 6.6\times10^{-3}$) and HRK4 ($h_{\max}^{\mathrm{HRK4}} \cong 4.4\times10^{-3}$) are stable for longer time steps. The main instability source stems from the fast oscillations triggered by the potential setting the bond length, Eq. (20). However, different from the other algorithms where only one force evaluation per integration step is needed, SHO and HRK4 require two and four force evaluations per integration step, respectively; i.e., they are computationally slower. It must be also noted that different algorithms involve the evaluation of different numbers of random variables. Even if the force evaluation is one crucial part of each integration step, the evaluation of random variables by using reliable routines[50,59] may affect the performance of the integration routines for small systems.[38] As outlined in Sec. II, the number of stochastic variables is different for each algorithm: SHO, BBK, and Li require two random variables, whereas SVV, SLO, and HRK4 involve only one random variable. As it will be evident in Sec. IV B, no systematic differences are evidenced at short h values. Increasing the h value leads to instabilities, which are related with the integration scheme of the different algorithms.

2. Stabilization of the deterministic part

In the low friction regime, of interest here, the effective integration carried out by both the stochastic and the deterministic parts of the algorithm is crucial. Augmenting symplectic algorithms by adding stochastic and dissipative terms seem to be a viable way to achieve this purpose.[38,39] Here, we want to clarify the improvement in the stabilization of the algorithms due to their stochastic part.

It is possible to classify the algorithms in two groups: Li, and BBK, stochastic VV, come from the VV splitting Eq. (10), whereas SLO and SHO are a stochastic derivation of the PV splitting which exchanges the definitions of the Liouvillians L_1 and L_2 with respect to the VV one; HRK4, instead, is a symplectic derivation of the standard Hamiltonian Runge-Kutta schemes.[60,61] It is known that the deterministic VV splitting is better at small time steps, but, on increasing the time step, it becomes unstable more quickly than the deterministic PV splitting.[47] For this reason it is usually thought that PV factorization is better than VV one for developing algorithms in order to reach longer time steps.[62,63]

Another approach to stabilize integration schemes with long time steps was pointed out by several authors, who suggested the use of a Langevin coupling as a device to dump numerical resonances associated with symplectic

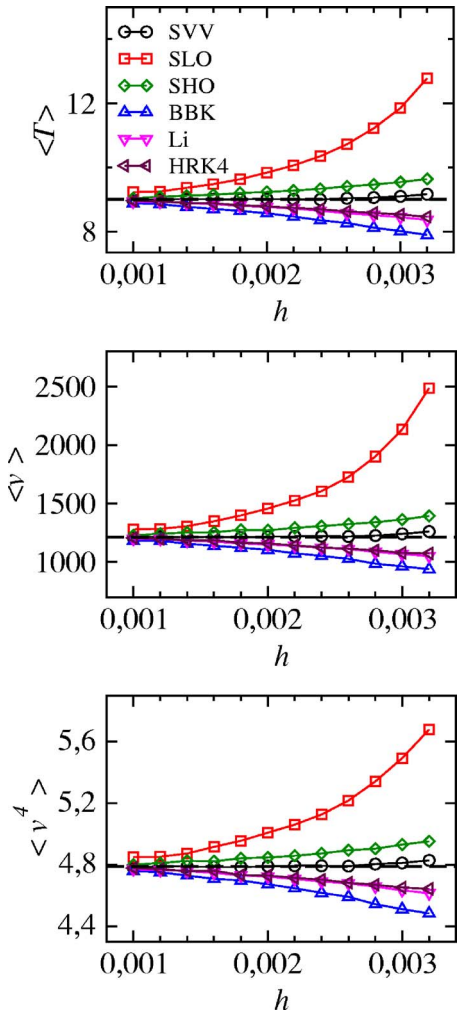

FIG. 1. (Color online) The average temperature $\langle T\rangle$ and the first and fourth moments of the probability distribution of the modulus of the velocity $\langle v\rangle$ and $\langle v^4\rangle$, respectively. The superimposed dashed lines are the expected theoretical values. The absence of error bars from the present figure and the next ones implies that they are smaller than the symbol size.

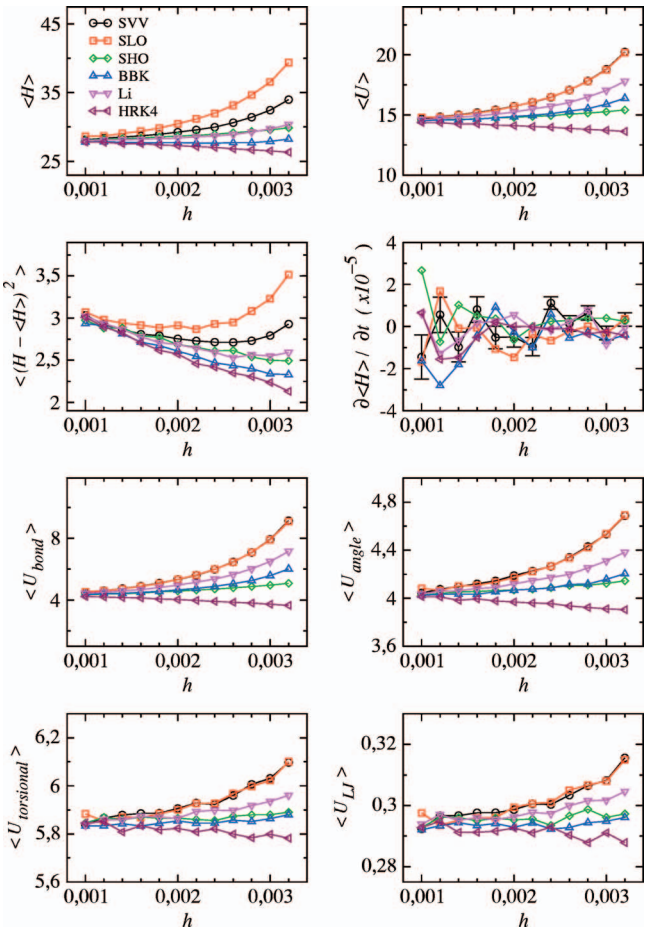

FIG. 2. (Color) Dependence of the different energy terms on the integration step h. The quantities which are plotted are the average values per monomer of the overall energy $\langle H\rangle$, the overall potential energy $\langle U\rangle$, and its different contributions, $\langle U_{\text{bond}}\rangle$ [Eq. (20)], $\langle U_{\text{angle}}\rangle$ [Eq. (21)], and $\langle U_{\text{torsion}}\rangle$ [Eq. (22)], as well the Lennard-Jones average energy $\langle U_{\text{LJ}}\rangle$ [Eq. (23)]. The total energy fluctuations and the energy drift $\partial\langle H\rangle/\partial t$ were also plotted.

algorithms.[62–67] In order to illustrate the Langevin stabilization, let us compare the SVV and SLO algorithms both employing one single stochastic variable. The comparison is made, as usual, by evaluating the different kinetic and potential energy terms; see Figs. 1 and 2 to be discussed in greater detail in Sec. IV B 1. It is seen that the SVV algorithm is less stable as far as the potential energy is concerned, but the kinetic energy is more stable. On the other hand, SLO, derived by a PV splitting, performs as SVV to evaluate the potential energy, but it is worse as far as the kinetic energy is concerned. The total energy, as evaluated by SVV, is more stable. These findings suggest that the conclusions drawn for the deterministic algorithms (see above) cannot be extended to the stochastic counterparts in a straightforward way.

B. Detailed aspects

1. Energy conservation

The energy conservation is an important goal in MD simulations and the symplectic algorithms are designed to achieve this target. Figure 2 plots the average values of the overall energy $\langle H\rangle$, the overall potential energy $\langle U\rangle$, and its different contributions, $\langle U_{\text{bond}}\rangle$ [Eq. (20)], $\langle U_{\text{angle}}\rangle$ [Eq. (21)], and $\langle U_{\text{torsion}}\rangle$ [Eq. (22)], as well as the Lennard-Jones average energy $\langle U_{\text{LJ}}\rangle$ [Eq. (23)]. The total energy fluctuations and the energy drift $\partial\langle H\rangle/\partial t$ were also plotted. Monitoring the different components of the total energy is useful in order to search the sources of instabilities since each potential term has a characteristic time scale. From this respect, Fig. 2 shows that the energy contributions involving faster degrees of freedom, i.e., the bond potential, are more critical. Figure 2 shows that BBK is the most accurate in its stability region ($h<3.4\times10^{-3}$). SHO and Li have comparable performances. As we noted above, SHO is the most stable algorithm together with HRK4. However, the latter, a fourth-order algorithm, does not perform better than the lower-order algorithms, such as BBK or SHO. It is also seen that, even if SLO and Li get the probability distribution of the position with the same accuracy, the latter performs better. The limited accuracy of SVV is expected since it was developed for accurate evaluation of the velocity, see Sec. IV B 3. Table II summarizes the above remarks.

2. Structural properties

All the algorithms under study, apart from SVV, were designed to give accurate evaluations of the position distri-

TABLE II. Summary of the performances of the algorithms under study. Apart from the last three rows, the entries denote the ranks of the algorithms. The superscript denotes the sign of the positive/negative deviation from the best value. τ_X is the time needed by the X algorithm to complete one integration step. h_{max} is the integration time step above which the algorithm is unstable.

	Algorithm					
Quantity	SHO	BBK	HRK4	Li	SVV	SLO
$\langle H \rangle$	2^+	1^+	2^-	2^+	3^+	4^+
$\langle U \rangle$	1^+	2^+	2^-	3^+	4^+	4^+
$\langle U_{LJ} \rangle$	1^+	1^+	2^-	2^+	3^+	3^+
Bond distance	1^+	2^+	1^-	3^+	4^+	4^+
Bond angle	1^+	2^+	1^-	3^+	4^+	4^+
Torsional angle	1^+	1^+	2^-	2^+	3^+	3^+
Velocity	2^+	3^-	2^-	2^-	1^+	4^+
τ_X/τ_{SVV}	2	1	4	1	1	1
$h_{max} \times 10^3$	6.6	3.4	4.4	3.4	3.4	3.4

butions. To test the related performances, we computed the first two moments of the distribution of the bond length (Fig. 3) and the first four moments of the distributions of both the bond angle (Fig. 4) and the torsional angle (Fig. 5). Higher moments of the bond length distribution were found to be negligibly small. The evaluation of these structural properties evidences the better accuracy of SHO with respect to the other ones. HRK4 performs better than BBK. It must be noted that SVV and SLO, which have comparable low accuracy to evaluate both the potential energies (see Fig. 2) and the even moments of the distribution of the torsional angle (see Fig. 5), are rather different when both the bond length (Fig. 3) and the bond angle (Fig. 4) are considered. In fact, SLO underperforms the evaluation of the average values of both quantities but it has better accuracy for their higher moments, whereas SVV behaves in the opposite way. Table II summarizes the above remarks.

3. Temperature and moments of the velocity distribution

Finally, we compared the performances of the different algorithms when they have to evaluate the molecular velocity or related quantities such as the temperature. The results are presented in Fig. 1. The distribution of the modulus of the velocity is found to be Maxwellian in shape with first and fourth moments given by,[68]

$$\langle v \rangle = 2\sqrt{2T/\pi},$$

$$\langle v^4 \rangle = 15T^2.$$

As expected, SVV algorithm is quite accurate to evaluate both the temperature and the moments of the velocity distribution. SHO, Li, and HRK4 exhibit similar accuracy. From this respect, they are better than BBK.

Table II summarizes the above remarks.

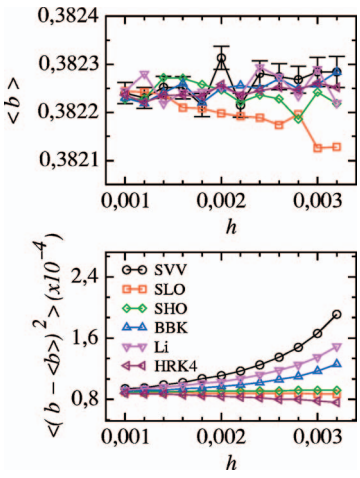

FIG. 3. (Color) The first two moments of the bond length distribution.

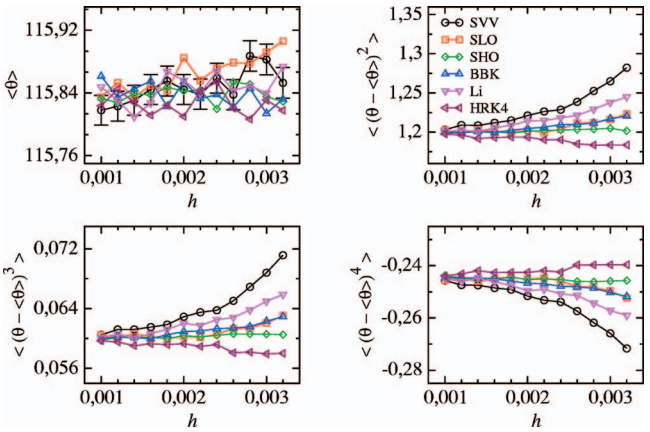

FIG. 4. (Color) The first four moments of the bond angle distribution.

V. CONCLUSIONS

Different algorithms were compared to test the accuracy, the numerical efficiency, and the stability of the MD simulations of a polymer solution with long integration time steps. HRK4 and the quasisymplectic SHO algorithm are the most stable. This feature is ensured by their greater accuracy to evaluate the position and, consequently, to follow the fast dynamics. However, the numerical efficiency is affected, especially for HRK4, by the increased number of evaluations of the forces per integration step. Differently, the popular

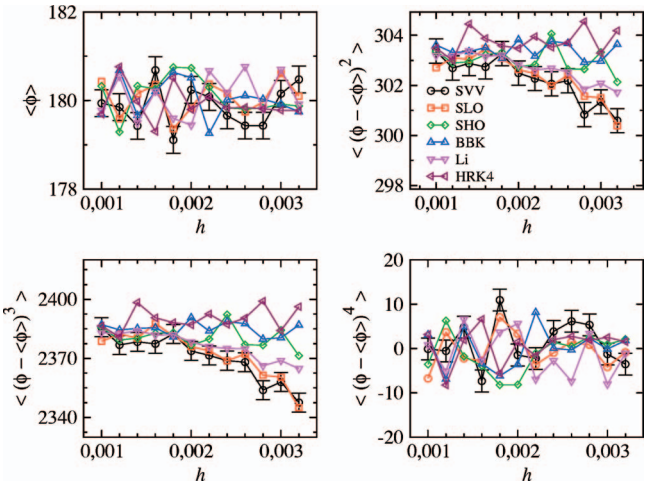

FIG. 5. (Color) The first four moments of the torsional angle distribution.

BBK algorithm is found to be faster with limited accuracy loss but with poorer stability. The dependence of the accuracy on the quantity to be evaluated was noted. As shown by Table II, the SHO algorithm exhibits better overall accuracy and stability than the other ones. As far as the computing efficiency is concerned, it also compares well with the faster (but less stable) BBK algorithm.

ACKNOWLEDGMENT

Financial support from MIUR within the PRIN project "Aging, fluctuation and response in out-of-equilibrium glassy systems" is gratefully acknowledged.

[1] P. G. de Gennes, *Scaling Concept in Polymer Physics* (Cornell University Press, Ithaca, NY, 1979).
[2] M. Doi and S. F. Edwards, *The Theory of Polymer Dynamics* (Clarendon, Oxford, 1988).
[3] G. Strobl, *The Physics of Polymers* (Springer, New York, 1997).
[4] A. Yu. Grosberg and A. R. Khokholov, *Giant Molecules* (Academic, New York, 1996).
[5] J. Baschnagel and F. Varnik, J. Phys.: Condens. Matter **17**, R851 (2005).
[6] L. R. Dodd, T. D. Boone, and D. N. Theodorou, Mol. Phys. **78**, 961 (1993).
[7] J. I. Siepmann and D. Frenkel, Mol. Phys. **75**, 59 (1992).
[8] J. J. de Pablo, M. Laso, and U. W. Suter, J. Chem. Phys. **96**, 2395 (1992).
[9] A. Uhlherr, Macromolecules **33**, 1351 (2000).
[10] P. V. K. Pant and D. N. Theodorou, Macromolecules **28**, 7224 (1995).
[11] V. G. Mavrantzas, T. D. Boone, E. Zervopoulou, and D. N. Theodorou, Macromolecules **32**, 5072 (1999).
[12] V. G. Mavrantzas and D. N. Theodorou, Comput. Theor. Polym. Sci. **10**, 1 (2000).
[13] S. Geyler, T. Pakula, and J. Reiter, J. Chem. Phys. **92**, 2676 (1990).
[14] S. Balijepalli and G. C. Rutledge, J. Chem. Phys. **109**, 6523 (1998).
[15] N. C. Karayiannis, V. G. Mavrantzas, and D. N. Theodorou, Phys. Rev. Lett. **88**, 105503 (2002).
[16] V. A. Harmandaris, V. G. Mavrantzas, D. N. Theodorou, M. Kröger, J. Ramirez, H. C. Ottinger, and D. Vlassopoulos, Macromolecules **36**, 1376 (2003).
[17] P. Minary, M. E. Tuckerman, and G. J. Martyna, Phys. Rev. Lett. **93**, 150201 (2004).
[18] D. Janežič, M. Praprotnik, and F. Merzel, J. Chem. Phys. **122**, 174101 (2005).
[19] M. Praprotnik and D. Janežič, J. Chem. Phys. **122**, 174102 (2005).
[20] M. Praprotnik and D. Janežič, J. Chem. Phys. **122**, 174103 (2005).
[21] M. Praprotnik, D. Janežič, and J. Mavri, J. Phys. Chem. A **108**, 11056 (2004).
[22] H. M. Chun, C. E. Padilla, D. N. Chin *et al.*, J. Comput. Chem. **21**, 159 (2000).
[23] A. Malevanets and R. Kapral, J. Chem. Phys. **112**, 7260 (2000).
[24] E. Villa, A. Balaeff, and K. Schulten, Proc. Natl. Acad. Sci. U.S.A. **102**, 6783 (2005).
[25] L. Delle Site, C. F. Abrams, A. Alavi, and K. Kremer, Phys. Rev. Lett. **89**, 156103 (2002).
[26] M. Praprotnik, L. Delle Site, and K. Kremer, J. Chem. Phys. **123**, 224106 (2005).
[27] A. Laio, J. V. Vondele, and U. Roethlisberger, J. Chem. Phys. **116**, 6941 (2002).
[28] G. S. Grest and K. Kremer, Phys. Rev. A **33**, 3628 (1986).
[29] K. Kremer and G. S. Grest, J. Chem. Phys. **92**, 5057 (1990).
[30] S. Nose, Prog. Theor. Phys. Suppl. **103**, 1 (1991).
[31] W. Hoover, Phys. Rev. A **31**, 1695 (1985).
[32] T. Schneider and E. Stoll, Phys. Rev. B **17**, 1302 (1978).
[33] J. N. Bright, D. J. Evans, and D. Searles, J. Chem. Phys. **122**, 194106 (2005).
[34] H. Mori, Prog. Theor. Phys. **34**, 399 (1965).
[35] R. Zwanzig, J. Stat. Phys. **9**, 215 (1973).
[36] B. Mishra and T. Schlick, J. Chem. Phys. **105**, 299 (1996).
[37] R. I. Cukier and M. Morillo, J. Chem. Phys. **123**, 234908 (2005).
[38] R. Mannella, Phys. Rev. E **69**, 041107 (2004).
[39] R. Mannella, SIAM J. Sci. Comput. (USA) **27**, 2121 (2006).
[40] H. Yoshida, Phys. Lett. A **150**, 262 (1990).
[41] R. D. Engle, R. D. Skeel, and M. Drees, J. Comput. Phys. **206**, 432 (2005).
[42] R. D. Skeel and D. J. Hardy, SIAM J. Sci. Comput. (USA) **23**, 1172 (2001).
[43] R. Mannella, Int. J. Mod. Phys. C **13**, 1177 (2002).
[44] K. Armitstead and G. Goldbeck-Wood, Adv. Polym. Sci. **100**, 219 (1992).
[45] E. Forest and R. D. Ruth, Physica D **43**, 105 (1990).
[46] H. F. Trotter, Proc. Am. Math. Soc. **10**, 545 (1959).
[47] M. Tuckerman, B. J. Berne, and G. J. Martyna, J. Chem. Phys. **97**, 1990 (1992).
[48] I. P. Omelyan, I. M. Mryglod, and R. Folk, Phys. Rev. E **65**, 056706 (2002).
[49] W. C. Swope, C. H. Andersen, P. H. Berens, and K. R. Wilson, J. Chem. Phys. **76**, 637 (1982).
[50] W. H. Press, A. S. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes: The Art of Scientific Computing* (Cambridge University Press, Cambridge, 2002).
[51] A. Brünger, C. B. Brooks, and M. Karplus, Chem. Phys. Lett. **105**, 495 (1984).
[52] M. P. Allen and D. J. Tildesley, *Computer Simulation of Liquids* (Clarendon, Oxford, 1987).
[53] L. Larini, A. Barbieri, D. Prevosto, P. A. Rolla, and D. Leporini, J. Phys.: Condens. Matter **17**, L199 (2005).
[54] L. Larini and D. Leporini, J. Chem. Phys. **123**, 144907 (2005).
[55] L. Larini, A. Barbieri, and D. Leporini, Physica A **364**, 183 (2006).
[56] W. Paul, D. Y. Yoon, and G. D. Smith, J. Chem. Phys. **103**, 1702 (1995).
[57] C. Liu and M. Muthukumar, J. Chem. Phys. **109**, 2536 (1998).
[58] M. Muthukumar and P. Welch, Polymer **41**, 8833 (2000).
[59] D. E. Knuth, *The Art of Computer Programming: Seminumerical Algorithms* (Addison Wesley, Reading, 1981), Vol. 2.
[60] I. P. Omelyan, I. M. Mryglod, and R. Folk, Comput. Phys. Commun. **151**, 273 (2003).
[61] M. Sofroniou and W. Oevel, SIAM (Soc. Ind. Appl. Math.) J. Numer. Anal. **34**, 2063 (1997).
[62] P. F. Batcho and T. Schlick, J. Chem. Phys. **115**, 4003 (2001).
[63] P. F. Batcho and T. Schlick, J. Chem. Phys. **115**, 4019 (2001).
[64] E. Barth and T. Schlick, J. Chem. Phys. **109**, 1617 (1998).
[65] E. Barth and T. Schlick, J. Chem. Phys. **109**, 1633 (1998).
[66] A. Sandu and T. Schlick, J. Comput. Phys. **151**, 74 (1999).
[67] J. A. Izaguirre, D. P. Catarello, J. M. Wozniak, and R. D. Skeel, J. Chem. Phys. **114**, 2090 (2001).
[68] A. Papoulis, *Probability, Random Variables, and Stochastic Processes* (McGraw-Hill, New York, 1984).