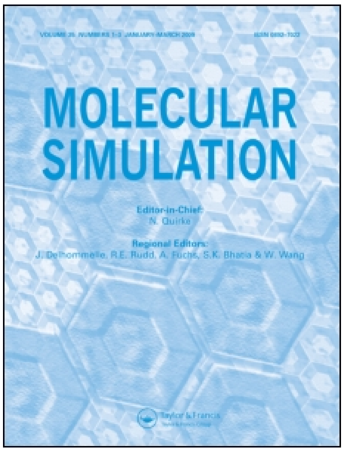
This article was downloaded by:
On: *14 January 2011*
Access details: *Access Details: Free Access*
Publisher *Taylor & Francis*
Informa Ltd Registered in England and Wales Registered Number: 1072954 Registered office: Mortimer House, 37-41 Mortimer Street, London W1T 3JH, UK



Molecular Simulation

Publication details, including instructions for authors and subscription information:
http://www.informaworld.com/smpp/title~content=t713644482

Non-Iterative Constraint Dynamics Using Velocity-Explicit Verlet Methods

Joseph T. Slusher[a,b]; Peter T. Cummings[a,b]
[a] Department of Chemical Engineering, University of Tennessee, Knoxville, TN [b] Chemical Technology Division, Oak Ridge National Laboratory, Oak Ridge, TN

To cite this Article Slusher, Joseph T. and Cummings, Peter T.(1996) 'Non-Iterative Constraint Dynamics Using Velocity-Explicit Verlet Methods', Molecular Simulation, 18: 4, 213 — 224
To link to this Article: DOI: 10.1080/08927029608022360
URL: http://dx.doi.org/10.1080/08927029608022360

PLEASE SCROLL DOWN FOR ARTICLE

Full terms and conditions of use: http://www.informaworld.com/terms-and-conditions-of-access.pdf

This article may be used for research, teaching and private study purposes. Any substantial or systematic reproduction, re-distribution, re-selling, loan or sub-licensing, systematic supply or distribution in any form to anyone is expressly forbidden.

The publisher does not give any warranty express or implied or make any representation that the contents will be complete or accurate or up to date. The accuracy of any instructions, formulae and drug doses should be independently verified with primary sources. The publisher shall not be liable for any loss, actions, claims, proceedings, demand or costs or damages whatsoever or howsoever caused arising directly or indirectly in connection with or arising out of the use of this material.

Molecular Simulation, 1996, Vol. 18, pp. 213–224
Reprints available directly from the publisher
Photocopying permitted by license only

NON-ITERATIVE CONSTRAINT DYNAMICS USING VELOCITY-EXPLICIT VERLET METHODS

JOSEPH T. SLUSHER and PETER T. CUMMINGS

Department of Chemical Engineering, University of Tennessee 419 Dougherty Engineering Bldg., Knoxville, TN 37996–2200 and Chemical Technology Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831–6268

(Received February 1996; accepted June 1996)

We describe non-iterative constraint dynamics algorithms for molecular simulation based on the velocity and Beeman [1, 3] forms of the Verlet algorithm. The methods are easily adapted for constant temperature molecular dynamics, and because they are non-iterative, should prove useful for large-scale simulations on parallel computers. Test simulations of liquid butane, water, and nitrogen are used to verify the accuracy, stability, and practical utility of the methods.

Keywords: Molecular dynamics simulation; constraint dynamics

1. INTRODUCTION

The implementation of holonomic constraints in molecular dynamics (MD) simulations has proven to be an effective method for simulating complex molecules. In the interest of facilitating efficient MD computations, constraint dynamics allows one to avoid following relatively fast internal motions, such as bond length vibrations, when these can be considered irrelevant to the properties under investigation. Although a number of numerical methods for determining the constraint forces have been presented, probably the most widely used are iterative algorithms known as SHAKE [4] and RATTLE [5]. By treating each constraint in succession and avoiding matrix inversion, these approaches have often been seen as being particularly attractive for molecules having a large number of constraints. Recent improvements to the convergence of the SHAKE algorithm have been reported which yield up to two-fold improvement over the original formulation [6]. These authors have also shown that special techniques for solving sparse matrices, applied to the so-called "matrix method" from which SHAKE is derived, can yield significant improvements over SHAKE, even when a large number of constraints are present. When implemented on parallel computer architectures, however, the iterative nature of the above methods can give rise to difficulties with load balancing as well as vectorisation and parallelization. To avoid these problems, a non-iterative method would therefore be highly advantageous (for a more thorough discussion see references [7] and [8]).

Two basic approaches have been taken in deriving non-iterative for constrained molecular dynamics. These begin with the use of undetermined multipliers λ_n to express the constrained equations of motion as

$$\dot{\mathbf{p}}_{i\alpha} = \mathbf{F}_{i\alpha} = \mathbf{F}^{N}_{i\alpha} + \mathbf{F}^{C}_{i\alpha}, \quad \mathbf{F}^{C}_{i\alpha} = \frac{1}{2}\sum_{n=1}^{n(\alpha)} \lambda_n \nabla_{\mathbf{r}_{i\alpha}} \chi_n, \tag{1}$$

where $\dot{\mathbf{p}}_{i\alpha}$ and $\mathbf{F}_{i\alpha}$ denote momenta of and force on site α of molecule i, N and C denote non-bonded and constraint forces, respectively, and $n(\alpha)$ is the number of constraints involving site α. The distance constraints are of the form

$$\chi_n(t) \equiv \mathbf{r}_n^2(t) - d_n^2 = 0, \quad n = 1, \ldots, M \tag{2}$$

where d_n is the fixed bond length between sites α and β, and $\mathbf{r}_n = \mathbf{r}_\beta - \mathbf{r}_\alpha$, where $\mathbf{r}_\gamma$ is the position of site γ. Differentiating, we also have

$$\dot{\chi}_n(t) = 2\mathbf{r}_n(t)\cdot\dot{\mathbf{r}}_n(t) = 0 \tag{3}$$

and

$$\ddot{\chi}_n(t) = 2\dot{\mathbf{r}}_n(t)^2 + 2\mathbf{r}_n(t)\cdot\ddot{\mathbf{r}}_n(t) = 0. \tag{4}$$

By eliminating the accelerations in (4) using the equations of motion (1), one can obtain a set of M linearly independent equations for the multipliers $\lambda(t)$. This was the approach taken by Edberg, Evans, and Morriss (EEM) who developed their non-iterative constraint method in light of Gauss's principle of least constraint [9]. A more formal development was presented by de Leeuw *et al.*, who began with the constrained Lagrangian equations of motion and, using (4) arrived at a similar method for solving for the forces of constraint [10]. They went further to derive Hamiltonian constrained equations of motion, and, using Equation (3) an alternative set of equations for the Lagrange multipliers.

The above prescriptions enable one to solve for the Lagrange multipliers exactly and straightforwardly. On the other hand, because they utilize the differential forms of the constraint condition (3) and (4), in which the constraint distances do not appear explicitly, they require that error accumulation be corrected for periodically during the simulation [9,10,11,12]. This complication led to the alternative approach taken by Yoneya, Berendsen, and Hirasawa (YBH), [8] who sought to develop a *self-correcting* non-iterative constraint method within the framework of the Verlet and leap-frog algorithms. Examining a Taylor expansion of the constraint condition,

$$\chi_n(t+h) + O(h^3) = \chi_n(t) + h\,\dot{\chi}_n(t) + (1/2)h^2\,\ddot{\chi}_n(t) \tag{5}$$

and noting that this equation is the same as that solved in the EEM method but with additional terms to suppress error accumulation, they obtained the following expression for $\chi_n(t+h)$ by using the Verlet algorithm in (2) and neglecting terms non-linear in λ:

$$\chi_n(t+h) = [2\mathbf{r}_n(t) - \mathbf{r}_n(t-h)]^2 - d_n^2 + 2h^2[2\mathbf{r}_n(t) - \mathbf{r}_n(t-h)] \cdot \ddot{\mathbf{r}}_n(t) + O(h^4) = 0. \tag{6}$$

This equation is the same as that solved in the linearized first step of the matrix method, so the above is equivalent to the matrix method if errors of $O(h^4)$ in the constraint condition are neglected. A set of M linear equations for $\lambda_n(t)$ are then obtained by inserting the equations of motion (1) in the form of $\ddot{\mathbf{r}}_n(t)$ into (6). They also used the leap-frog algorithim to obtain a similar equation for $\chi_n(t+h)$; However in this case the resulting expression is accurate only to $O(h^2)$. Whereas the error in the constraint condition can only be affected by changing the time-step size, the main advantage of the YBH approach compared to the earlier work is that correction for numerical drift is unnecessary. Both (6) and the analogous leap-frog expression contain the constraint distances, and furthermore, when positions are updated to $t+h$, they are satisfied at $t+2h$; Together these conditions lead to a self-correcting algorithm.

2. METHOD

With regard to numerical error accumulation, the YBH approach is attractive, but it is not entirely satisfactory for constant temperature simulations due to the fact that either the velocities are not explicit in the algorithm (Verlet), or are not available at the same time as the positions (leap-frog). Intuitively, it should also be possible to arrive at an equation for $\chi_n(t+h)$ similar to the above based on a velocity-explicit form of the Verlet algorithm such as the velocity Verlet method

$$\mathbf{r}(t+h) = \mathbf{r}(t) + h\dot{\mathbf{r}}(t) + \frac{h^2}{2}\mathbf{f}\,[\mathbf{r}(t)] \tag{7}$$

$$\dot{\mathbf{r}}(t+h) = \dot{\mathbf{r}}(t) + \frac{h}{2}[\mathbf{f}\,[\mathbf{r}(t)] + \mathbf{f}\,[\mathbf{r}(t+h)]], \tag{8}$$

where $\mathbf{f} = \mathbf{F}/m$ and m is the mass. If we assume this can be implemented in a constraint dynamics scheme, then the $\mathbf{f}[\mathbf{r}(t)]$ calculated at the previous time step will already include the correct constraint forces necessary to ensure the position updates satisfy (2). The apparent problem arises in evaluating (8) when we consider that in order to evaluate $\mathbf{f}[\mathbf{r}(t+h)]$ such that $\mathbf{r}(t+2h)$ satisfy (2), we first need $\dot{\mathbf{r}}(t+h)$ before $\mathbf{r}(t+h)$ can be calculated. The solution to this problem in Andersen's RATTLE algorithm is to use (3) to obtain approximate values of the constraint forces which can be used in the velocity updates [5]. With RATTLE, the constraint forces must be recalculated at the next time step, although Palmer recently showed that it is possible to apply the velocity Verlet algorithm to constraint dynamics with only one constraint force evaluation per time step, in a method similar to SHAKE [13]. This is accomplished by rewriting $\mathbf{r}(t+2h)$ (and thus $\chi_n(t+2h)$) in terms of quantities that are available at time t, and using this in (2), solving for the updated constraint forces which will then not only allow the velocity update to $t+h$, but also ensure that $\mathbf{r}(t+2h)$ satisfy (2).

YBH did not apply their method to the velocity Verlet algorithm because of the apparent necessity of evaluating constraint forces twice per step. However, by writing the position update (7) in terms of the *previous* velocity, we can avoid this problem. Substituting (8) into (7) we obtain

$$\mathbf{r}_n(t+h) = \mathbf{r}_n(t) + h\dot{\mathbf{r}}_n(t-h) + \frac{h^2}{2}\ddot{\mathbf{r}}_n(t-h) + h^2\ddot{\mathbf{r}}_n(t) \tag{9}$$

and using this in (2) yields an expression for $\chi_n(t+h)$ linear in $\lambda(t)$ that is third-order accurate:

$$\chi_n(t+h) = r_n^2(t) - d_n^2 + 2h\mathbf{r}_n(t)\cdot\dot{\mathbf{r}}_n(t-h) + h^2\dot{\mathbf{r}}_n^2(t-h) +$$

$$h^2\{\mathbf{r}_n(t) + h\dot{\mathbf{r}}_n(t-h)\}\cdot\{2\ddot{\mathbf{r}}_n(t) + \ddot{\mathbf{r}}_n(t-h)\} + O(h^4) = 0 \tag{10}$$

In the velocity Verlet algorithm, this equation can be applied immediately after the position update and force evaluation, and in this way allows the same constraint forces to be used in both the position and velocity updates. Substituting the constrained equations of motion for $\ddot{\mathbf{r}}_n(t)$ in (10) gives a rank M matrix which can be solved using, *e.g.*, routines from LAPACK based on efficient machine-dependent BLAS routines [2]. For the simplest case, a diatomic molecule with equations of motion

$$m_\alpha\dot{\mathbf{r}}_\alpha = p_\alpha, \quad \dot{p}_1 = F_1 - \lambda\mathbf{r}_{12}, \quad \dot{p}_2 = \mathbf{F}_2 + \lambda\mathbf{r}_{12}, \quad \alpha = 1, 2 \tag{11}$$

substituting (11) into (10) yields the following expression for the unknown Lagrange multiplier λ:

$$\lambda(t) = -[\mathbf{r}_{12}^2(t) - d_{12}^2 + 2h\mathbf{r}_{12}(t)\cdot\dot{\mathbf{r}}_{12}(t-h) + \dot{\mathbf{r}}_{12}^2(t-h) +$$

$$h^2\{\mathbf{r}_{12}(t) + h\dot{\mathbf{r}}_{12}(t-h)\}\cdot\{\ddot{\mathbf{r}}_{12}(t-h) +$$

$$2\mathbf{F}'_{12}(t)\}]/2h^2\mu_{12}\mathbf{r}_{12}(t)\cdot\{\mathbf{r}_{12}(t) + h\dot{\mathbf{r}}_{12}(t-h)\} \tag{12}$$

where $\mathbf{F}'_{12} = \mathbf{F}_2/m_2 - \mathbf{F}_1/m_1$ and $\mu_{12} = 1/m_1 + 1/m_2$.

We can follow the same procedure to obtain a non-iterative matrix method based on the Beeman variant of the Verlet algorithm [3]

$$\mathbf{r}(t+h) = \mathbf{r}(t) + h\dot{\mathbf{r}}(t) + \frac{2}{3}h^2\mathbf{f}[\mathbf{r}(t)] - \frac{1}{6}h^2\mathbf{f}[\mathbf{r}(t-h)] \tag{13}$$

$$\dot{\mathbf{r}}(t+h) = \dot{\mathbf{r}}(t) + \frac{1}{3}h\mathbf{f}[\mathbf{r}(t+h) + \frac{5}{6}h\mathbf{f}[\mathbf{r}(t)] - \frac{1}{6}h\mathbf{f}[\mathbf{r}(t-h)]. \tag{14}$$

and we obtain

$$\chi_n(t+h) = \mathbf{r}_n^2(t) - d_n^2 + 2h\mathbf{r}_n(t)\cdot\dot{\mathbf{r}}_n(t-h) + h^2\dot{\mathbf{r}}_n^2(t-h) +$$

$$h^2\{\mathbf{r}_n(t) + h\dot{\mathbf{r}}_n(t-h)\}\cdot\{2\ddot{\mathbf{r}}_n(t) + \frac{4}{3}\ddot{\mathbf{r}}_n(t-h) - \frac{1}{3}\ddot{\mathbf{r}}_n(t-2h)\} + O(h^4) = 0. \tag{15}$$

Substituting Taylor expansions for the terms at $t-h$ and $t-2h$, and neglecting terms containing $\dddot{\mathbf{r}}_n(t)$, which are not considered in the algorithm, all three equations for $\chi_n(t+h)$ (6), (10) and (15) are equivalent to (5) expanded to one more term. Furthermore, since the Verlet, velocity Verlet, and Beeman algorithms differ only in the velocity estimator, we can expect all three constraint methods to generate identical trajectories.

3. TEST SIMULATIONS

We performed NVE and NVT simulations of both rigid and flexible molecules in order to test the algorithms for relatively simple systems. For butane the model of Ryckaert and Bellemans [15] was used, which constraints both the bond lengths and bond angles, giving a total of 5 constraints. For the simulation of N_2 we used the model of Cheung and Powles [16]. Test simulations of SPC/E water [18] were run in order to evaluate the algorithm in a practical case where long-ranged forces are involved and the molecule contains disparate site masses. All simulations were performed at liquid state conditions, with spherical cutoff at $L/2$ (L being the length of the side of the simulation cube), and without long-range correction for the Lennard-Jones interactions. The water simulations were run at near ambient conditions using Ewald summation to handle the long-range electrostatic interactions, with the Ewald parameter [14] $\kappa = 6.0/L$, real-space truncation at $L/2$, and a maximum lattice vector length of $5L$. Additional simulation details for the above are given in Tables I and II, and the matrix equations solved for water and butane are given in the appendix.

The NVT simulations were carried out using the Nosé-Hoover approach based on an atomic thermostat, *i.e.*, the momentum equations

TABLE I Summary of Constraint Errors for Water, Butane, and N_2

subst.	*ensbl.*	N	*time* (*ps*)	h (*fs*)	$T(K)$	$\rho(g/cm^3)$	$\langle\chi^2\rangle^{1/2a}$	$\lvert\chi^{max}\rvert^a$	$\chi^2_{max}{}^b$
water	NVE	108	10	1.0	286.0	0.997	2.2×10^{-5}	5.0×10^{-5}	2.8×10^{-11}
water	NVT	108	10	1.0	300.6	0.997	2.3×10^{-5}	1.2×10^{-4}	1.6×10^{-10}
butane	NVE	32	100	2.0	316.8	0.583	3.8×10^{-7}	6.8×10^{-7}	2.0×10^{-15}
butane	NVT	32	100	2.0	291.6	0.583	9.8×10^{-7}	5.0×10^{-6}	1.1×10^{-13}
N_2	NVE	108	30	2.3	70.5	0.828	4.1×10^{-7}	8.0×10^{-6}	5.3×10^{-13}
N_2	NVT	108	30	2.3	71.6	0.828	3.6×10^{-7}	1.1×10^{-5}	1.0×10^{-12}

[a] Units of Å^2.
[b] dimensionless with respect to σ.

TABLE II Total and Configurational Energies for Water and Butane

subst.	*ensemble.*	N	*time* (*ps*)	h (*fs*)	$T(K)$	$\rho(g/cm^3)$	$\langle(E-\bar{E})^2\rangle^{1/2a}$	$U^C(KJ/mol)$
water	NVE	108	10	1.0	286.0	0.997	0.04	-42.0
water	NVT	108	10	1.0	300.6	0.997	–	$-41.0(-41.1)^b$
butane	NVE	108	10	2.0	294.2	0.583	0.003	-18.4
butane	NVT	32	100	2.0	291.6	0.583	–	$-13.0(-13.7)^c$

[a] dimensionless with respect to ε.
[b] ref. [19]
[c] ref. [9], $N = 27$

become

$$\dot{\mathbf{p}}_{i\alpha} = \mathbf{F}_{i\alpha} - \xi \mathbf{p}_{i\alpha}, \tag{16}$$

where ξ is a thermostating variable that changes with time according to [17]

$$\dot{\xi} = v_T^2 \left(\frac{T(t)}{T_{\text{set}}} - 1 \right), \tag{17}$$

with T_{set} being the set-point temperature, $T(t)$ the instantaneous temperature based on the momenta of all sites, and v_T^2 the thermostating rate set equal to 1000.

In order to check the actual order of the methods for bond constraint errors, the root mean squared (RMS) error of χ (averaged over all constraints for 100 time-steps) was calculated as a function of the time-step h. The results in Figure 1 confirm that both the velocity Verlet-based algorithm and Beeman-based algorithms conserve χ with errors proportional to h^4. These results are comparable with those of YBH for butane using the Verlet- and leap-frog-based constraint methods, also shown in the figure. Although the error in χ for the water simulations was $O(h^4)$ (slope = 4.27), the magnitude of the errors is greater compared to the butane and N_2 results, reflecting the relative stiffness of the equations for the water simulations. In Table I we present the simulation details for longer runs using typical time-step sizes. These results show that the effect of the thermostat, in most cases, is to slightly increase the RMS bond length errors. However, this effect is not large and the maximum constraint errors are of the same order as typically used in SHAKE [8]. When non-dimensionalized with respect to the Lennard-Jones size parameter σ,

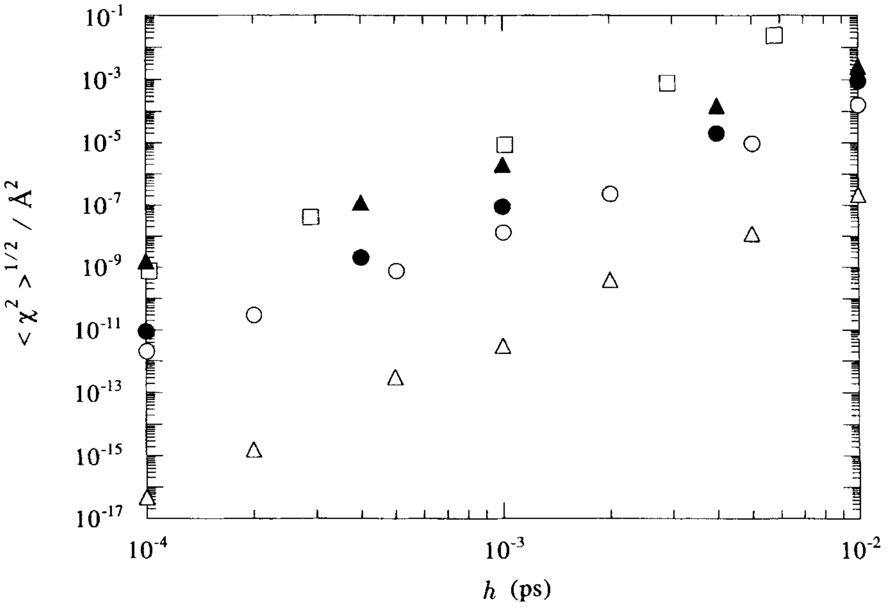


FIGURE 1 Variation in root mean squared constraint error with time-step size. (○), butane results with Beeman-based method; (□), SPC/E water results with Beeman-based method; (△), N_2 using velocity-Verlet. Included for comparison are the YBH results for butane using the Verlet-based (●), and leap-frog-based method (▲).

the maximum constraint errors are smaller than the tolerances used by EEM [9].

The validity of the methods can be further assessed by checking the conservation of total energy and through comparison of thermodynamic properties such as the configurational energy with literature values. For the water and butane simulations using the Beeman method, the RMS % energy fluctuations were acceptably low: <0.1% and <0.02% for water and butane, respectively. The simulation details are given in Table II, where we present results for two NVT simulations of water and butane with setpoint temperatures of 300.6 and 291 K corresponding to state points at which literature results are available. The calculated configurational energies for these simulations agree well with published values at the same state points.

Although the above results suggest that the algorithms are sufficiently accurate compared to other constraint methods, they do not of themselves constitute a proof that the dynamics are correct. One way to check this is to compare dynamical results from simulations of diatomic molecules, for which the constraint forces can be determined "exactly", with those obtained using

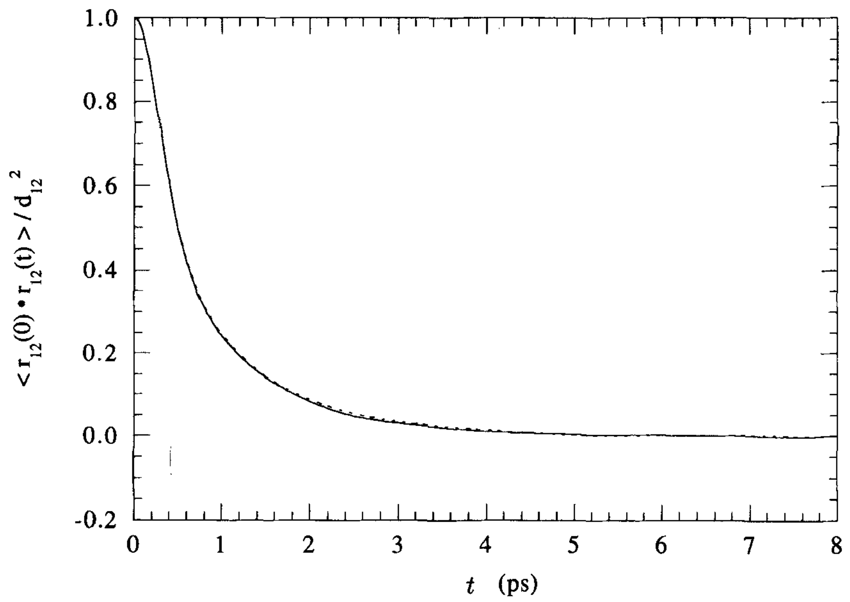


FIGURE 2 Comparison of reorientation autocorrelation for N_2 using "exact" constraint algorithm (–) with the velocity-Verlet-based non-iterative constraint method (···). The simulations were run at constant NVE for 460 ps, with $\rho = 0.828$ g/cm^3, $T = 70.3$ K.

the above algorithms. In Figure 2 we show a plot of a reorientational correlation function, defined as $\langle \mathbf{r}_{12}(0) \cdot \mathbf{r}_{12}(t) \rangle$, for model liquid N_2. From this we see that the dynamical results using, in this case the velocity Verlet-based method, are nearly indistinguishable from those of the exact algorithm.

4. CONCLUSIONS

In this work, we have extended the YHB algorithms for non-iterative constrained molecular dynamics to include the velocity explicit Verlet algorithms. Like the Verlet- and leap-frog-based methods, these are self-correcting and in addition, allow for a more satisfactory treatment of the velocities. They are easily adapted for isothermal simulations through the use of, *e.g.*, a Nosé-Hoover thermostat without significantly affecting the bond constraint errors. We found the methods to be both accurate and stable during test simulations of N_2, butane, and SPC/E water. Due to their non-iterative nature, they should be more suited for large-scale simulations on parallel computers. Although we do not report precise timings of the code, the algorithms presented are of comparable speed to the Verlet- and leap-frog-based methods, in which the constraint portion of the code was about 5 times faster than SHAKE for test simulations of small molecules [8].

Acknowledgements

The authors are grateful for financial assistance from the Office of Basic Energy Sciences, U.S. Department of Energy. JTS thanks the reviewer for detailed criticism, and A. A. Chialvo and A. Baranyai for helpful advice.

6. APPENDIX

In solving $\mathbf{A}\lambda = \mathbf{b}$ for the Lagrange multipliers $\lambda(t)$, only the vector $\mathbf{b}$ differs between the velocity Verlet- and Beeman-based methods. For water we have 3 constraints applied between sites 1 and 2, 1 and 3, and 2 and 3. Numbering these consecutively and denoting $\mathbf{r}_{12}(t)$ as $\mathbf{r}_1$, etc., gives

$$A = \begin{bmatrix} \mu_1\mathbf{r}_1\cdot\mathbf{g}_1 & \frac{\mathbf{r}_2\cdot\mathbf{g}_1}{m_1} & -\frac{\mathbf{r}_3\cdot\mathbf{g}_1}{m_2} \\ \frac{\mathbf{r}_1\cdot\mathbf{g}_2}{m_1} & \mu_2\mathbf{r}_2\cdot\mathbf{g}_2 & \frac{\mathbf{r}_3\cdot\mathbf{g}_2}{m_3} \\ -\frac{\mathbf{r}_1\cdot\mathbf{g}_3}{m_2} & \frac{\mathbf{r}_2\cdot\mathbf{g}_3}{m_3} & \mu_3\mathbf{r}_3\cdot\mathbf{g}_3 \end{bmatrix},$$

where $\mathbf{g}_k = \mathbf{r}_k + \dot{\mathbf{r}}_k(t-h)$.

For butane, with constraints 1–5 between sites 1 and 2, 1 and 3, 2 and 3, 2 and 4, and 3 and 4 we have

$$\mathbf{A} = \begin{bmatrix} \mu_1\mathbf{r}_1\cdot\mathbf{g}_1 & 0 & -\frac{\mathbf{r}_3\cdot\mathbf{g}_1}{m_2} & -\frac{\mathbf{r}_4\cdot\mathbf{g}_1}{m_2} & 0 \\ \frac{\mathbf{r}_1\cdot\mathbf{g}_2}{m_1} & \mu_2\mathbf{r}_2\cdot g_2 & \frac{\mathbf{r}_3\cdot\mathbf{g}_2}{m_3} & 0 & -\frac{\mathbf{r}_5\cdot\mathbf{g}_2}{m_3} \\ -\frac{\mathbf{r}_1\cdot\mathbf{g}_3}{m_2} & \frac{\mathbf{r}_2\cdot\mathbf{g}_3}{m_3} & \mu_3\mathbf{r}_3\cdot\mathbf{g}_3 & -\frac{\mathbf{r}_4\cdot\mathbf{g}_3}{m_2} & -\frac{\mathbf{r}_5\cdot\mathbf{g}_3}{m_3} \\ -\frac{\mathbf{r}_1\cdot\mathbf{g}_4}{m_2} & 0 & \frac{\mathbf{r}_3\cdot\mathbf{g}_4}{m_2} & \mu_4\mathbf{r}_4\cdot\mathbf{g}_4 & \frac{\mathbf{r}_5\cdot\mathbf{g}_4}{m_4} \\ 0 & -\frac{\mathbf{r}_2\cdot\mathbf{g}_5}{m_3} & -\frac{\mathbf{r}_3\cdot\mathbf{g}_5}{m_3} & \frac{\mathbf{r}_4\cdot\mathbf{g}_5}{m_4} & \mu_5\mathbf{r}_5\cdot\mathbf{g}_5 \end{bmatrix}.$$

The elements of the vector $\mathbf{b}$ are, for the velocity Verlet method,

$$b_k = -\frac{1}{2h^2}[\mathbf{r}_k^2 - d_k^2 + 2h\mathbf{r}_k\cdot\dot{\mathbf{r}}_k(t-h) + h^2\dot{\mathbf{r}}_k^2(t-h)$$

$$+ h^2\{\mathbf{r}_k + h\dot{\mathbf{r}}_k(t-h)\}\cdot\{2\mathbf{F}'_k + \ddot{\mathbf{r}}_k(t-h)\}]$$

and for the Beeman method

$$b_k = -\frac{1}{2h^2}[\mathbf{r}_k^2 - d_k^2 + 2h\mathbf{r}_k\cdot\dot{\mathbf{r}}_k(t-h) + h^2\dot{\mathbf{r}}_k^2(t-h) + h^2\{\mathbf{r}_k + h\dot{\mathbf{r}}_k(t-h)\}\cdot$$

$$\{2\mathbf{F}'_k + \frac{4}{3}\ddot{\mathbf{r}}_k(t-h) - \frac{1}{3}\ddot{\mathbf{r}}_k(t-2h)\}]\cdot$$

Note that the above expressions contain quantities which are unknown at the start of the simulation, *i.e.* at $t-h$ and $t-2h$. In practice, these can be set to zero at startup until they have been accumulated.

References

[1] Swope, W. C., Andersen, H. C., Berens, P. H. and Wilson, K. R. (1982) "A computer simulation method for the calculation of equilibrium constants for the formation of physical clusters of molecules: Application to small water clusters", *J. Chem. Phys.*, **76**, 637.

[2] Anderson, E., Bai, Z., Bischof, C., Demmel, J., Dongarra, J., Du Croz, J., Greenbaum, A., Hammarling, S., McKenney, A., Ostrouchov, S. and Sorensen, D. (1995) *LAPACK User's Guide, Release 2.0*, Siam Publications, Phila.

[3] Beeman, D. (1976) "Some multistep methods for use in molecular dynamics calculations", *J. Comp. Phys.*, **20**, 130.

[4] Ryckaert, J. P., Ciccotti, G. and Berendsen, H. J. C. (1977) "Numerical integration of the Cartesian equations of motion of a system with constraints: Molecular dynamics of *n*-alkanes", *J. Comp. Phys.*, **23**, 327.

[5] Anderson, H. C. (1983) "Rattle: A 'velocity' version of the shake algorithm for molecular dynamics calculations", *J. Comp. Phys.*, **52**, 24.

[6] Barth, E., Kuczera, K., Leimkuhler, B and Skeel, R. D. (1995) "Algorithms for constrained molecular dynamics", *J. Comp. Chem.*, **16**, 1192.

[7] Muller-Plathe, F. and Brown, D. (1991) "Multi-colour algorithms in molecular simulation: Vectorisation and parallelisation of internal forces and constraints", *Comput. Phys. Commun.*, **64**, 7.

[8] Yoneya, M., Berendsen, H. J. C. and Hirasawa, K. (1994) "A non-iterative matrix method for constraint molecular dynamics simulations", *Mol. Sim.*, **13**, 395.

[9] Edberg, R., Evans, D. J. and Morriss, G. P. (1986) "Constrained molecular dynamics simulations of liquid alkanes with a new algorithm", *J. Chem. Phys.*, **84**, 6933.

[10] de Leeuw, S. W., Perram, J. W. and Petersen, H. G. (1990) "Hamilton's equations for constrained dynamical systems", *J. Stat. Phys.*, **61**, 1203.

[11] Baranyai, A. and Evans, D. J. (1990) "A new algorithm for constrained dynamics simulations of liquid benzene and naphthalene", *Mol. Phys.*, **70**, 53.

[12] Daivis, P. J., Evans, D. J. and Morriss, G. P. (1992) "Computer simulation study of the comparative rheology of branched and linear alkanes", *J. Chem. Phys.*, **97**, 616.

[13] Palmer, B. J. (1993) "Direct application of SHAKE to the velocity Verlet Algorithm", *J. Comp. Phys.*, **104**, 470.

[14] Allen, M. P. and Tildesley, D. J. (1987) *Computer Simulation of Liquids*, Oxford University Press, New York.

[15] Ryckaert, J.-P. and Bellemans, A. (1978) "Molecular dynamics of liquid alkanes", *Discuss. Faraday Soc.*, **66** 95.

[16] Cheung, P. S. Y. and Powles, J. G. (1975) "The properties of liquid nitrogen IV. A computer simulation", *Mol. Phys.*, **30**, 921.
[17] Melchionna, S., Ciccotti, G. and Holian, B. (1993) "Hoover *NPT* dynamics for systems varying in shape and size", *Mol. Phys.*, **78**, 533.
[18] Berendsen, H. J. C., Grigera, J. R. and Straatsma, T. P. (1987) "The missing term in effective pair potentials", *J. Phys. Chem.*, **91**, 6269.
[19] Heyes, D. M. (1994) "Physical properties of liquid water by molecular dynamics simulations", *J. Chem. Soc. Faraday Trans.*, **90**, 3039.