Symplectic Numerical Integrators in Constrained Hamiltonian Systems [1]

Benedict J. Leimkuhler
Department of Mathematics
(University of Kansas)
405 Snow Hall
Lawrence, KS 66045, U.S.A.

Robert D. Skeel
Department of Computer Science
(University of Illinois)
1304 West Springfield Avenue
Urbana, IL 61801 U.S.A.

Keywords: Störmer method, Verlet method, constrained dynamics, constrained Hamiltonian system, SHAKE, RATTLE, symplectic discretization, canonical method, molecular dynamics simulation.

AMS(MOS) Subject Classifications: 65L20, 65L05

[1] The first author was supported in part by National Science Foundation Grant DMS 92 055538; the second author was supported in part by National Science Foundation Grant DMS 90 15533

Running Head: Symplectic Integrators

Address Correspondence to:

Benedict J. Leimkuhler

Department of Mathematics

405 Snow Hall

The University of Kansas

Lawrence, KS 66045, U.S.A.

leimkuhl@math.ukans.edu

Abstract:

Recent work reported in the literature suggests that for the long-time integration of Hamiltonian dynamical systems one should use methods that preserve the symplectic (or canonical) structure of the flow. Here we investigate the symplecticness of numerical integrators for *constrained dynamics*, such as occur in molecular dynamics when bond lengths are made rigid in order to overcome stepsize limitations due to the highest frequencies. This leads to a constrained Hamiltonian system of smaller dimension. Previous work has shown that it is possible to have methods which are symplectic on the constraint manifold in phase space. Here it is shown that the very popular Verlet method with **SHAKE**-type constraints is equivalent to the same method with **RATTLE**-type constraints and that the latter is symplectic and time reversible. (This assumes that the iteration is carried to convergence.) We also demonstrate the global convergence of the Verlet scheme in the presence of **SHAKE**–type and **RATTLE**–type constraints. We perform numerical experiments to compare these methods with the second-order backward differentiation method, commonly recommended for ordinary differential equations (ODEs) with constraints.

1. Introduction. Vibrational Newtonian models used in molecular dynamics simulations lead to Hamiltonian systems of ordinary differential equations of the form

$$M\dot{q} = p \tag{1}$$

$$\dot{p} = -\nabla_q V(q) \tag{2}$$

where $q, p \in \mathbf{R}^n$ are, respectively, positions and momenta of atoms of the molecule which are regarded as point masses, M is a positive definite (typically diagonal) mass matrix and $V : \mathbf{R}^n \to \mathbf{R}$ is a potential function. In [22, 27], rigid bonds were incorporated into molecular models to improve efficiency. Fixing bond lengths and bond angles in the vibrational model results in holonomic constraints of the form $g(q) = 0$ and leads to constrained dynamical equations (Lagrangian equations of the first kind):

$$M\dot{q} = p \tag{3}$$

$$\dot{p} = -\nabla_q V(q) + g'(q)^t \lambda \tag{4}$$

$$g(q) = 0 \tag{5}$$

(see, e.g., Hildebrand [15]).

A simple two-step discretization was used by Verlet [28] to solve (1)–(2) and it remains the most popular discretization scheme for unconstrained equations. In [22], a direct numerical integration scheme (**SHAKE**) based on the Verlet method and preserving the constraint relationships was presented for (3)–(5). This scheme was later adapted by Andersen [3] into an alternative velocity-level formulation that preserves certain additional invariants; this is the basis of the **RATTLE** algorithm.

The flow of a Hamiltonian system like (1)–(2) possesses an important *symplectic* geometric structure [4]. Briefly, the sum of the areas of the projections of an oriented two-dimensional surface in phase space onto the $q_i p_i$ coordinate planes is unchanged under the flow. Much recent research has gone into developing symplectic numerical discretization schemes that inherit the symplectic structure of the original system. It has been observed in numerical experiments [20] that symplectic methods with fixed stepsize possess better long-term stability properties than nonsymplectic methods. In [21] it was demonstrated that the Verlet method is symplectic, and further work (recently surveyed by Sanz-Serna [24]) has uncovered a variety of symplectic discretization schemes.

The symplectic integration of the constrained equations (3)–(5) was treated in [16] via symplectic parameterization of the constraint manifold and by methods based on Dirac's theory of weak invariants. These methods lead to unconstrained Hamiltonian systems which can be handled by direct application of symplectic methods, but the new Hamiltonians are nonseparable, meaning that they cannot be written in the form $H(q,p) = T(p) + V(q)$, and hence are not amenable to discretization via explicit symplectic methods like the Verlet scheme. This appears to rule out parameterization and weak-invariant methods for the computationally intensive molecular dynamics application.

In this paper, we consider direct symplectic numerical discretizations of the constrained equations (3)–(5). We show that the Verlet methods with **SHAKE**–type and **RATTLE**–type constraint algorithms yield identical solutions for the positions at meshpoints and at half-steps in the velocity. Both methods preserve the wedge product, although **SHAKE** is not, strictly speaking, a symplectic method, as the meshpoint velocities are not tangent to the constraint manifold defined by (5). Verlet with either **SHAKE**–type or **RATTLE**–type constraints is time reversible, an important feature for difference schemes [7]. We demonstrate that both methods are reducible to a certain discretization of an unconstrained system of differential equation, from which a global convergence result follows. Other numerical methods for constrained differential equations based on backward differentiation formulas ("BDF" methods) are in use for engineering problems in constrained form; we compare **RATTLE** in numerical experiments with a second-order BDF method, and find the symplectic scheme the clear winner.

2. Symplectic Maps and Symplectic Discretization Schemes. For full rank g', we refer to the set $\mathcal{M} = \{(q,p) \mid g(q) = 0, g'(q)M^{-1}p = 0\}$ as the "solution manifold" associated with (3)–(5). (This definition is justified by observing that the equations (3)–(5) define a *hidden constraint* $g'(q)M^{-1}p = 0$ which must be satisfied by initial values (q_0, p_0) in order to guarantee existence of smooth solutions.)

To define a natural "symplectic structure" [4] on $\mathcal{M}$, we first parameterize the manifold locally in $2n-2m$ variables, say positions $\delta \in \mathbf{R}^{n-m}$ and momenta $\theta \in \mathbf{R}^{n-m}$. This can be done in such a way that rewriting the differential equations (3)-(5) as equations in the parameters results in an unconstrained Hamiltonian system, meaning that the 2-form $d\delta \wedge d\theta$ is preserved by the reduced flow. Then it can be shown [16] that with p and q restricted to $\mathcal{M}$, $d\delta \wedge d\theta = dq \wedge dp$. The flow (on $\mathcal{M} \subset \mathbf{R}^{2n}$) of (3)-(5 preserves the 2-form $dq \wedge dp$. This motivates the following definition: Let $(Q, P) = \phi(q, p)$ be a differentiable mapping from $\mathcal{M}$ into itself, ϕ is said to be a *symplectic mapping* if it preserves the restriction to $\mathcal{M}$ of the symplectic form in $\mathbf{R}^{2n}$ i.e. if $dQ \wedge dP = dq \wedge dp$.

A *one-step* method for (3)–(5) is defined as a mapping $\psi_h : \mathcal{M} \to \mathcal{M}$ on a parameter h that takes (q_n, p_n) into (q_{n+1}, p_{n+1}). One of the simplest symplectic discretizations (i.e., for which ψ_h is a symplectic mapping) for integrating the unconstrained equations (1)–(2) is the Verlet scheme:[2]

$$q_{n+1} - 2q_n + q_{n-1} = -h^2 M^{-1} \nabla_q V(q_n) \tag{6}$$

Verlet defined $p_n = M(q_{n+1} - q_{n-1})/2h$. These equations can be rewritten in the following velocity formulation:

$$q_{n+1} = q_n + hM^{-1} p_{n+1/2} \tag{7}$$

$$p_{n+1/2} = p_n - (h/2)\nabla_q V(q_n) \tag{8}$$

$$p_{n+1} = p_{n+1/2} - (h/2)\nabla_q V(q_{n+1}) \tag{9}$$

Here (q_n, p_n) represents an approximation to the solution $(q(t_n), p(t_n))$ at time t_n, and the constant stepsize h is just the difference of any two successive time points: $t_{n+1} = t_n + h$.

[2] This scheme is sometimes referred to by numerical analysts as Störmer's rule.

Although (7)–(9) and (6) are mathematically equivalent, it is known (see, e.g., Hairer, Nørsett, and Wanner [14, p. 430]) that (6) has an instability with respect to rounding errors which is not present in leap-frog or in (7)–(9), hence the velocity formulation is to be preferred.

The method (7)–(9) is second-order accurate in time, meaning that on a fixed interval $[0, T]$, if we define the error at the nth time level $t_n = nh \leq T$ in the solution computed with stepsize h as $e_n^h = \|(q_n, p_n) - (q(t_n), p(t_n))\|$, then $e_n^h = O(h^2)$, uniformly, i.e., there are constants $C > 0$ and $h_0 > 0$ such that

$$e_n^h \leq Ch^2$$

whenever $h < h_0$ and $0 \leq n \leq N = T/h$.

A simple linear coordinate transformation allows us to restrict attention in the sequel to the case $M = I$. Setting $q = M^{-1/2}\hat{q}$ and $p = M^{1/2}\hat{p}$ in (1)–(2) or in (3)–(5), and setting $V(q) = \hat{V}(M^{1/2}q)$ and $g(q) = \hat{g}(M^{1/2}q)$, we find that the form of the equations is unchanged except that in the new system the mass matrix is I. All of the results of this paper apply equally to cases with M any positive definite matrix (so that a positive definite square root $M^{1/2}$ is defined).

3. Direct Symplectic Discretization of Constrained Systems. A popular method for adapting the Verlet method to handle bond-length and bond-angle constraints is the **SHAKE** algorithm [22]. If the algorithm, as described in [2], is iterated to convergence, it can be expressed in our notation (with $M = I$) as

$$q_{n+1} = 2q_n - q_{n-1} - h^2\nabla_q V(q_n) + h^2 g'(q_n)^t \lambda_n \tag{10}$$

$$g(q_{n+1}) = 0 \tag{11}$$

Setting $p_{n+1/2} = (q_{n+1} - q_n)/h$ yields the leap-frog form:

$$q_{n+1} = q_n + hp_{n+1/2} \tag{12}$$

$$p_{n+1/2} = p_{n-1/2} - h\nabla_q V(q_n) + hg'(q_n)^t \lambda_n \tag{13}$$

$$g(q_{n+1}) = 0 \tag{14}$$

The local error occurring after one step with (12)–(14) is $O(h^3)$. If we further define $p_n = (q_{n+1} - q_{n-1})/(2h)$, we obtain

$$\begin{aligned} q_{n+1} &= q_n + hp_{n+1/2} && (15)\\ p_{n+1/2} &= p_n - (h/2)\nabla_q V(q_n) + (h/2)g'(q_n)^t\lambda_n && (16)\\ g(q_{n+1}) &= 0 && (17)\\ p_{n+1} &= p_{n+1/2} - (h/2)\nabla_q V(q_{n+1}) + (h/2)g'(q_{n+1})^t\lambda_{n+1} && (18) \end{aligned}$$

For convenience, and to distinguish our formulation from other possible formulations of **SHAKE**, we refer to (15)–(18) as **VS** (for *velocity-level SHAKE*). VS cannot be a symplectic method as we have defined it above since, although $g(q_n) = 0$ at every grid point, the hidden constraint will typically fail to be satisfied: $g'(q_n)M^{-1}p_n \neq 0$, even when the starting values lie in $\mathcal{M}$. On the other hand we can get another perspective by viewing (15)–(18) as a one-step mapping in $\mathcal{M}_0 = \{(q,p)|g(q) = 0\}$. Here the differentials obey

$$\begin{aligned} dq_{n+1} &= dq_n + hdp_{n+1/2} && (19)\\ dp_{n+1/2} &= dp_n - (h/2)d\nabla_q V(q_n) + (h/2)d(g'(q_n)^t\lambda_n) && (20)\\ g'(q_{n+1})dq_{n+1} &= 0 && (21)\\ dp_{n+1} &= dp_{n+1/2} - (h/2)d\nabla_q V(q_{n+1}) + (h/2)d(g'(q_{n+1})^t\lambda_{n+1}) && (22) \end{aligned}$$

Letting the Hessian of V be denoted by V'', we see that $d\nabla_q V(q_n) = V''(q_n)dq_n$. Taking the wedge product of differentials at the end of a step, we have

$$\begin{aligned} dq_{n+1} \wedge dp_{n+1} &= dq_{n+1} \wedge (dp_{n+1/2} - (h/2)V''(q_{n+1})dq_{n+1})\\ &= dq_{n+1} \wedge dp_{n+1/2} - (h/2)dq_{n+1} \wedge V''(q_{n+1})dq_{n+1} && (23)\\ &\quad +(h/2)dq_{n+1} \wedge d(g'(q_{n+1})^t\lambda_{n+1}) \end{aligned}$$

The second term in (23) can be eliminated by use of the following lemma which follows from the skew-symmetry of the wedge product and from the fact that for a matrix B of appropriate dimensions, $du \wedge (Bdv) = (B^t du) \wedge dv$.

Lemma 3.1. *Let du be an arbitrary differential in $\mathbf{R}^n$, and let A be any $n \times n$ real symmetric matrix then $du \wedge (Adu) = 0$.*

Another lemma allows us to eliminate the third term in (23):

Lemma 3.2. $dq_n \wedge d(g'(q_n)^t \lambda_n) = 0$

Proof:

$$dq_n \wedge d(g'(q_n)^t \lambda_n) = dq_n \wedge g'(q_n)^t d\lambda_n + \sum_{i=0}^{m} \lambda_n^i dq_n \wedge \Gamma_n^i dq_n$$

where the components of λ_n have been indexed by a superscript and Γ_n^i is the Hessian of the ith constraint function.

Now $dq_n \wedge g'(q_n)^t d\lambda_n = g'(q_n) dq_n \wedge d\lambda_n = 0$, since $g(q_n) = 0 \Rightarrow g'(q_n) dq_n = 0$. Each of the terms of the summation can be eliminated by Lemma 1, proving Lemma 2. □

Applying the lemmas in (23), we arrive at

$$dq_{n+1} \wedge dp_{n+1} = dq_{n+1} \wedge dp_{n+1/2}$$

Next, from (19), we have

$$\begin{aligned} dq_{n+1} \wedge dp_{n+1/2} &= (dq_n + hdp_{n+1/2}) \wedge dp_{n+1/2} \\ &= dq_n \wedge dp_{n+1/2} \\ &= dq_n \wedge (dp_n - (h/2) d\nabla_q V(q_n) + hd(g'(q_n)^t \lambda_n)) \end{aligned}$$

Applying both lemmas to simplify the latter expression, we have shown that

$$dq_{n+1} \wedge dp_{n+1} = dq_n \wedge dp_n$$

and it follows that (15)–(18) preserves the wedge product.

4. Velocity-Level Constraints: RATTLE. Although **VS** does not define a symplectic mapping, one can correct this deficiency by a simple device: if the momenta p_{n+1} are projected onto the hidden constraints, then the result is a one-step mapping that both (i) maps $\mathcal{M}$ into $\mathcal{M}$

and (ii) preserves the wedge product. The converged **RATTLE** algorithm [3] can be expressed as

$$
\begin{aligned}
q_{n+1} &= q_n + hp_{n+1/2} && (24)\\
p_{n+1/2} &= p_n - \frac{h}{2}\nabla_q V(q_n) + \frac{h}{2}g'(q_n)^t\lambda_n^{(\mathrm{r})} && (25)
\end{aligned}
$$

where $\lambda_n^{(\mathrm{r})}$ is chosen so that

$$g(q_{n+1}) = 0 \tag{26}$$

and

$$p_{n+1} = p_{n+1/2} - \frac{h}{2}\nabla_q V(q_{n+1}) + \frac{h}{2}g'(q_{n+1})^t\lambda_{n+1}^{(\mathrm{v})} \tag{27}$$

where $\lambda_{n+1}^{(\mathrm{v})}$ is chosen so that

$$g'(q_{n+1})p_{n+1} = 0 \tag{28}$$

We refer to (24)–(28) as a **VR** step. If we write

$$q_{n+1} = q_n + hp_{n+1/2}$$

then we find

$$p_{n+1/2} = p_{n-1/2} - h\nabla_q V(q_n) + \frac{h}{2}g'(q_n)^t(\lambda_n^{(\mathrm{r})} + \lambda_n^{(\mathrm{v})})$$

where $\lambda_n^{(\mathrm{v})}$ can be written explicitly in terms of p_n and $\nabla_q V(q_n)$. Here $\lambda_n^{(\mathrm{r})}$ is chosen so that

$$g(q_n + hp_{n+1/2}) = 0$$

we recognize that this is equivalent to the leap-frog variant of the **VS** method with $\lambda_n = \frac{1}{2}(\lambda_n^{(\mathrm{r})} + \lambda_n^{(\mathrm{v})})$, which is simply a change of variables for the unknown Lagrange multipliers.

Thus **VS**=**VR** at the half steps $t_{n+1/2}$, but **RATTLE** satisfies both position and velocity constraints at meshpoints.

To see that solutions generated by the **RATTLE** method at meshpoints preserve the wedge product, we note that, from (27),

$$dq_{n+1} \wedge dp_{n+1} = dq_{n+1} \wedge dp_{n+1/2} - \frac{h}{2} dq_{n+1} \wedge d\nabla_q V(q_{n+1}) + \frac{h}{2} dq_{n+1} \wedge dg'(q_{n+1})^t \lambda_{n+1}^{(\mathrm{v})}$$

by the reasoning of the previous section, the latter two terms here vanish, and then again using the arguments of the previous section, we have that

$$dq_{n+1} \wedge dp_{n+1/2} = dq_n \wedge dp_{n-1/2}$$

and it follows that

$$dq_{n+1} \wedge dp_{n+1} = dq_n \wedge dp_n$$

so the method is symplectic.

Note that **VR** can also be viewed as a method by which the **VS** steps are symplectically projected into $\mathcal{M}$, since we can rewrite the equations as

$$\begin{aligned}
&(29) & q_{n+1} &= q_n + hp_{n+1/2} \\
&(30) & p_{n+1/2} &= p_n - (h/2)\nabla_q V(q_n) + (h/2) g'(q_n)^t \lambda_n \\
&(31) & g(q_{n+1}) &= 0 \\
&(32) & p_{n+1} &= (I - \mathcal{H}(q_{n+1}))(p_{n+1/2} - (h/2)\nabla_q V(q_{n+1}) + (h/2) g'(q_{n+1})^t \lambda_{n+1})
\end{aligned}$$

where $\mathcal{H}(q) = g'(q)^t (g'(q) g'(q)^t)^{-1} g'(q)$ is a projector matrix.

5. Global Convergence of the SHAKE and RATTLE Methods. (15)–(18), with or without the endpoint projection defines an approximation of (3)–(5) which is *locally* $O(h^3)$. This means that starting from the same initial point, the exact and approximate solutions will differ after a single time step of size h by an amount that tends to zero like the third power of h; however, this fact alone does not imply global second-order convergence of the method as the classical (unconstrained) ODE discretization theory does not carry over directly to the constrained case.

Noting that both bond angle and bond length type constraints are quadratic greatly simplifies the discussion. As we have seen, the fact that the constraint (5) is identically satisfied along solutions implies that the equation $g'(q)p = 0$ is also satisfied along solutions. Differentiating the latter equation with respect to time yields

$$g'(q)\dot{p} + g''[p,p] = 0$$

where $g''[.,.]$ represents the tensor second derivative on two arguments. Because the constraints are assumed quadratic, the expression $g''[p,p]$ is independent of q. Substituting for $\dot{p}$ from (4), we get

$$g'(q)(-\nabla_q V(q) + g'(q)^t \lambda) + g''[p,p] = 0$$

We can solve this equation for $\lambda = \lambda(q,p)$. Reintroducing this expression in (3)–(4) results in

$$\dot{q} = p \tag{33}$$

$$\dot{p} = -(I - \mathcal{H})\nabla_q V(q) - g'(q)^t (g'(q) g'(q)^t)^{-1} g''[p,p] \tag{34}$$

This is an (unconstrained) ODE system which preserves the hidden constraint as an integral invariant. It is probably not computationally efficient to formulate and directly integrate (33)–(34) numerically, and, moreover, this *underlying ODE* is not typically Hamiltonian.[3] However, the following theorem shows that if the equations of **SHAKE** and **RATTLE** are solved sufficiently accurately, then they are equivalent to the direct discretization of (33)-(34) by a certain numerical scheme.

THEOREM 5.1. *Neglecting rounding errors, the* **VR** *and* **VS** *methods are step-by-step equivalent to the discretization of (33)-(34) by a convergent second-order ODE method. Hence both the* **VR** *and* **VS** *schemes are globally second-order convergent.* The proof is given in Appendix A. In the study of differential-algebraic equations, one frequently finds that rounding errors or other errors introduced due to inexact solution of the nonlinear equations may lead to instabilities in the numerical solution. We are currently investigating this question.

[3] For the construction of Hamiltonian *underlying ODEs*, see [16].

6. Nonsymplectic Discretizations of Constrained Problems. The numerical solution of general *differential-algebraic equations* (DAEs), including problems of the form (3)–(5), has been an important recent topic in scientific computing. General methods, as well as specialized schemes for solving the equations of motion of multibody mechanical systems are surveyed in, e.g., [5, 9]. Among multistep methods, the methods based on *backward differentiation formulas* ("BDF methods") have been shown to converge with fixed and variable steps for constrained problems [17, 11]. Applying the k-step fixed-stepsize BDF formulas to (3)–(5) results in equations:

$$\sum_{i=0}^{k} \alpha_i q_{n+1-i} = hM^{-1} p_{n+1} \tag{35}$$

$$\sum_{i=0}^{k} \alpha_i p_{n+1-i} = -h\nabla_q V(q_{n+1}) + hg'(q_{n+1})^t \lambda_{n+1} \tag{36}$$

$$0 = g(q_{n+1}) \tag{37}$$

One step of this method requires substantially more computational work than **VS** or **RATTLE**, since the method treats both g and $\nabla_q V$ implicitly. On the other hand, more efficient, semi-implicit implementations [9, 18] are in use in multibody dynamics. The BDF formulas are known to be highly stable methods (they are suitable for "stiff" ordinary differential equations [10]), and one expects some dissipation of energy when they are used. The BDF methods are popular as solution methods for general systems of differential-algebraic equations, and they form the basis for numerical integration in the research code DASSL.

7. Numerical Comparisons Among the Methods. We implemented **VS** and **VR** iterations and the second-order BDF method (35)–(37) (with $k = 2$). The methods were applied to a simple test problem consisting of a set of six nodes connected together by springs with spring constant κ as shown in Figure 1. The coordinates of the i'th node in the figure are labelled (q_{2i-1}, q_{2i}), and the corresponding momenta are (p_{2i-1}, p_{2i}). The problem was simulated on the time interval $[0, 10]$ with various choices of the stepsize h. Although much more complicated model problems should eventually be treated, particularly in conjunction with the evaluation of various implementation strategies, the simple problem does illuminate important basic aspects of the methods discussed here.

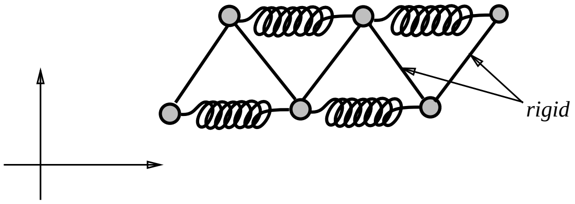

FIG. 1. *Test Problem.*

The Lagrangian equations (in cartesian coordinates) describing the problem of Figure 1 are in the form (3)–(5) with q and p taking values in $\mathbf{R}^{12}$, $M = I_{12}$ (the 12×12 identity), $V = \frac{5}{2} q^t K q$, where

$$K = \kappa \begin{bmatrix} I_2 & 0 & -I_2 & 0 & 0 & 0 \\ 0 & I_2 & 0 & -I_2 & 0 & 0 \\ -I_2 & 0 & 2I_2 & 0 & -I_2 & 0 \\ 0 & -I_2 & 0 & 2I_2 & 0 & -I_2 \\ 0 & 0 & -I_2 & 0 & I_2 & 0 \\ 0 & 0 & 0 & -I_2 & 0 & I_2 \end{bmatrix}$$

and

$$g(q) = \frac{1}{2} \begin{bmatrix} (q_1 - q_3)^2 + (q_2 - q_4)^2 - 1 \\ (q_3 - q_5)^2 + (q_4 - q_6)^2 - 1 \\ (q_5 - q_7)^2 + (q_6 - q_8)^2 - 1 \\ (q_7 - q_9)^2 + (q_8 - q_{10})^2 - 1 \\ (q_9 - q_{11})^2 + (q_{10} - q_{12})^2 - 1 \end{bmatrix}$$

7.1. Implementation. The efficient implementation of **SHAKE** or **RATTLE**-type constraints for large molecules requires that careful use be made of the available special (sparse) structure present in the constraints of typical molecular problems. We outline the simplified approach used here to obtain numerical comparisons of the underlying time-stepping schemes.

For the implementation of an implicit method, the first problem is to choose an explicit *predictor* that provides an initial guess for an iterative solution of the nonlinear equations. In our experiments, we used a quadratic interpolating polynomial passed through the previous solution values as a predictor.

The equations (15)–(18) can be rewritten as a nonlinear system for λ_n. We employed an iteration equivalent to an approximate Newton iteration on this nonlinear system. As a stopping criterion for the iteration, we have demanded that the change in the iterates and the normed residual of g be smaller than a certain prescribed tolerance *gtol* in 2-norm. Note that in order for **VR** or **VS** to be symplectic we must accurately solve the nonlinear equations.

In the case of **VR**, once q_{n+1} and $p_{n+1/2}$ p_{n+1} is computed by solving another linear system.

7.2. Velocity-Level Constraints in SHAKE. We first investigated the behavior of the residual of the velocity-level constraint $g'(q)p = 0$ when the **VS** method is used (of course, **RATTLE** satisfies this constraint to rounding error). Graphs of the velocity constraint residual for **VS** with $h = 0.01$ are shown in Figure 2. The figure illustrates the fact that, although error will be introduced in the velocity level when using **SHAKE**, these errors are not amplified from step to step. (This fact can also be demonstrated analytically.)

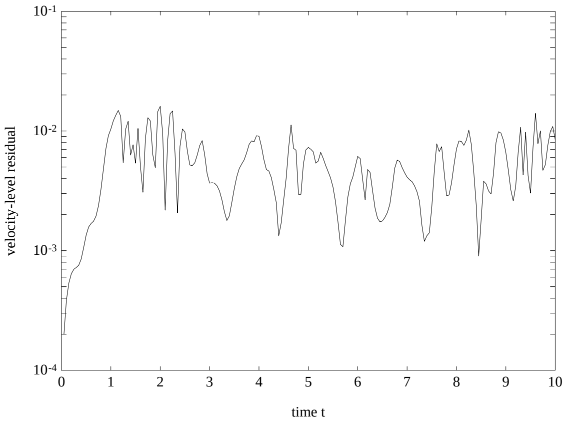

FIG. 2. *Residual of the Velocity-Level Constraint: VS Scheme.*

7.3. Comparison of RATTLE and BDF(2). We next compared the behavior of the **VR** and BDF methods. We computed the numerical solutions for various values of the stepsize, and compared with a baseline solution computed with a much smaller stepsize than those used in any of the other runs to obtain estimates for the numerical error. Figure 3, showing the endpoint numerical error (at $t = 10$) versus stepsize in log-log scale, indicates the clear superiority of the symplectic scheme in terms of accuracy for a given stepsize. In Figure 4 we have plotted the endpoint absolute energy error vs. the stepsize used, again in log-log scale, showing an even greater spread between the symplectic and nonsymplectic integrators.

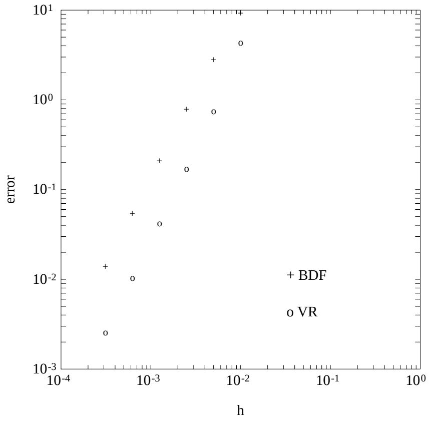

FIG. 3. *Comparison of Numerical Error: VR vs. BDF.*

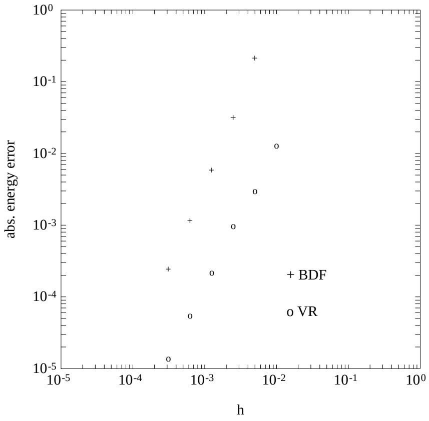

FIG. 4. *Comparison of Absolute Energy Error: VR vs. BDF.*

Finally, we considered the behavior over time of the energy error in **VR** and BDF(2) discretizations with an identical stepsize of $h = 0.01$. In Figure 5, we see that, even on a relatively long time interval of $[0, 100]$, the energy for **RATTLE** is approximately conserved in comparison with BDF.

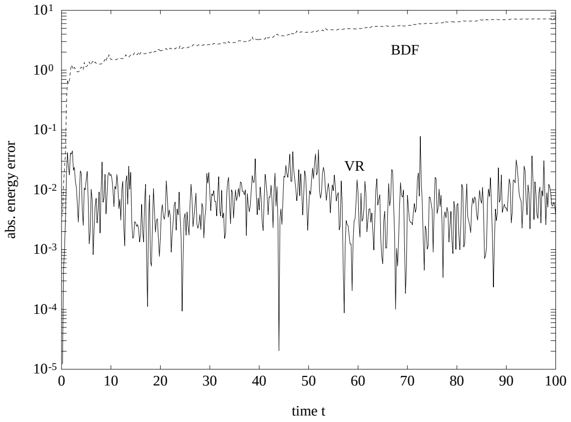

FIG. 5. *Conservation of Energy over Time: VR vs. BDF.*

7.4. A Lennard-Jones Model Problem. To get a more realistic model for the molecular dynamics problem, we looked at a small "molecule" consisting of a planar constraint chain of seven atoms in a Lennard-Jones (6-12) potential. Specifically, for two atoms at distance d, the potential between them was

$$\phi(d) = .1(d^{-12} - 2d^{-6})$$

The problem was used in [23] as a model for energy minimization. We started with an initial configuration at the global minimum of potential and applied vertical initial velocities of magnitude 0.25 in opposite directions at the ends of the chain. In this way, the linear momentum of the molecule center of mass is zero, and the whole structure spins in place. There are essentially two components of the motion: (1) a "rigid body" motion and (2) a vibration due to the Lennard-Jones potential. We integrated using the VR and BDF(2) schemes on $[0, 200]$ with a stepsize of $h = .1$ and a tolerance of 0.000001 for the nonlinear solver. Figure 6 compares the motion of the first atom of the chain as computed by the two integrators. After a very short interval, the BDF scheme has removed the vibrational components which are well resolved on the entire interval by the VR method. The energies are compared in Figure 7.

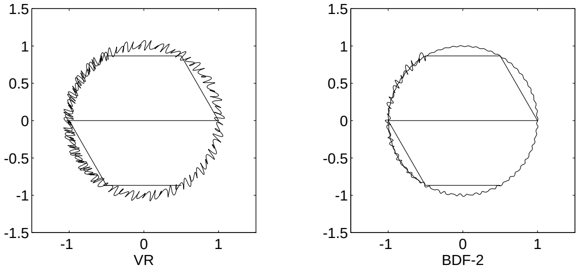

FIG. 6. *BDF2 and RATTLE Motion of One Atom on the time interval [0,200].*

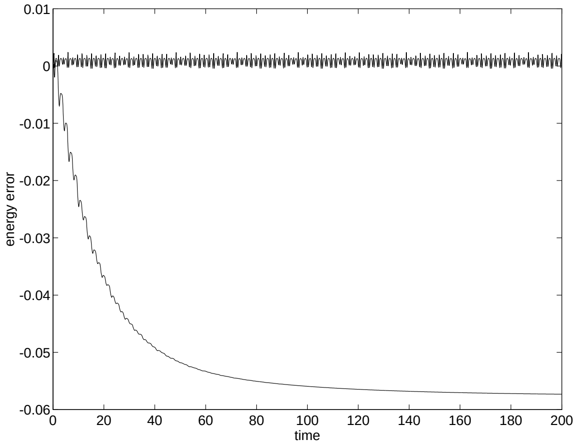

FIG. 7. *Energy for BDF-2 and RATTLE Solutions.*

For this example, we observed that the RATTLE scheme converged for larger stepsizes than the BDF method. The damping capability of the BDF family of integrators is sometimes exploited in order to "integrate over" fast *stiff* modes with large stepsizes, however, the artificial dissipation that these methods introduce is frequently inappropriate to the physical nature of the problem.

8. Conclusion. From our experience, if the constraint relationships are solved accurately enough (*gtol* is sufficiently small), then **VS** and **VR** produce equivalent results, although **VS** iterates do not satisfy the hidden constraints. Both methods will outperform BDF in terms of computational efficiency. Since the **VS** iteration can probably be implemented most efficiently of all, one suspects that the ideal method consists of a sequence of **VS** steps followed by a **VR**

step only when output is desired.

REFERENCES

[1] Abraham, R., and Marsden, J.E., *Foundations of Mechanics*, 2nd Ed., Benjamin Cummings Publishing Co., Inc., Reading, MA., 1978.

[2] Allen, M.P., and Tildesley, D.J.,*Computer Simulation of Liquids*, Oxford Science Publications, 1987.

[3] Andersen, H.C., Rattle: a 'velocity' version of the shake algorithm for molecular dynamics calculations, *J. Comp. Phys.*, 52:24-34, 1983.

[4] Arnold, V.I., *Mathematical Methods of Classical Mechanics*, Springer Graduate Texts in Mathematics No. 60, Springer-Verlag, NY, 1975.

[5] Brenan, K.E., Campbell, S.L. and Petzold, L.R., *Numerical Solution of Initial-Value Problems in Differential -Algebraic Equations*, North-Holland, 1989.

[6] Brooks, B., Bruccoleri, R., Olafson, B., States, D., Swaminathan, S., and Karplus, M., CHARMm: A program for macromolecular energy, minimization, and dynamics calculations, *J. Comp. Chem.*, 4:187-217, 1983.

[7] Buneman, O., Time-reversible difference procedures, *J. Comp. Phys.*, 1:517-535, 1967.

[8] Dirac, P.A.M., *Lectures on Quantum Mechanics*, Belfer Graduate School Monographs, No. 3, Yeshiva University, 1964.

[9] Eich, E., Führer, C., Leimkuhler, B., and Reich, S., Stabilization and Projection Methods for Multibody Dynamics, Report A281, Helsinki U. of Tech., 1990.

[10] Gear, C. W., Numerical Initial Value Problems in Ordinary Differential Equations, Prentice-Hall, 1971.

[11] Gear, C.W., Gupta, G.K., and Leimkuhler, B.J., Automatic integration of the Euler-Lagrange equations with constraints, *J. Comp. Appl. Math.*, 12&13:77–90, 1985.

[12] Golub, G.H., and Van Loan, C., *Matrix Computations*, 2nd. Ed., Johns Hopkins U. Press, 1989.

[13] Hairer, E., Lubich, Ch., Roche, M.: The numerical solution of differential-algebraic equations by Runge-Kutta methods. Lecture Notes in Mathematics, 1409 , New York, Berlin, Heidelberg: Springer 1989

[14] Hairer, E., Nørsett, S., and Wanner, G., *Solving Ordinary Differential Equations, I*, Springer-Verlag, 1987.

[15] Hildebrand, F.B., *Methods of Applied Mathematics*, 2nd Ed., Prentice-Hall, 1965.

[16] Leimkuhler, B., and Reich, S., Symplectic integration of constrained Hamiltonian systems, *Math. Comp.*, to appear.

[17] Lötstedt, P., and Petzold, L.R., Numerical solution of nonlinear differential equations with algebraic con-

straints I: convergence results for backward differentiation formulas, *Math. Comput.*, 46:491-516.

[18] Lubich, C., and Nowak, U., Tech. Report, Konrad-Zuse-Zentrum für Informationstechnik Berlin, 1992.

[19] März, R., On the numerical treatment of differential-algebraic equations, preprint, Humboldt-Universität zu Berlin, Germany.

[20] Okunbor,D., and Skeel, R.D., Explicit canonical methods for Hamiltonian systems, *Math. Comput. 59*, 1992, 439–455.

[21] Ruth, R., A canonical integration technique, *IEEE Trans. Nuc. Sci.*, 30:2669–2671, 1983.

[22] Ryckaert, J.P., Ciccotti, G., and Berendsen, H.J.C., Numerical integration of the cartesian equations of motion of a system with constraints: molecular dynamics of n-alkanes, *J. Comp. Phys.*, 23:327-341, 1977.

[23] Judson, R.S., Colvin, M.E., Meza, J.C., Huffer, A., and Gutierrez, D., Do intelligent configuration search techniques outperform random search for large molecules?, *Int. J. Quant. Chem.*, 44:277-290, 1992.

[24] Sanz-Serna, J.M. Symplectic integrators for Hamiltonian problems: An overview, *Acta Numerica 1*, 243–286, 1992.

[25] Sewell, M.J., and Roulstone, I., Anatomy of the Canonical Transformation, Scientific Paper No. 6, Short Range Forecasting Division, Meteorological Office, UK.

[26] Scovel, C., Symplectic numerical integration of Hamiltonian systems, in proceedings of *The Geometry of Hamiltonian Systems*, ed., T. Ratiu, Springer-Verlag, 463–496, 1992.

[27] Tobias, D.J. and Brooks C.L., Molecular dynamics with internal coordinate constraints, *J. Chem. Phys.*, 89:5115-5127, 1988.

[28] Verlet, L., Computer experiments on classical fluids. I. Thermodynamical properties of Lennard-Jones molecules, *Phys. Rev.*, 159:98-103, 1967.

[29] Winske, D. and Omidi, N., Hybrid codes: methods and applications, in *Lecture notes for the fourth international school for space simulation*, Kyoto, 1991.

Appendix A. Proof of Global Convergence of SHAKE and RATTLE Before proceeding with the proof of Theorem 5.1, consider the general unconstrained second-order system of ordinary differential equations

$$\begin{aligned} \dot{q} &= p \\ \dot{p} &= \phi(q, p) \end{aligned}$$

with smooth ϕ. These equations can be discretized by the following scheme

$$q_{n+1} = q_n + hp_{n+1/2} \tag{38}$$

$$p_{n+1/2} = p_{n-1/2} + \frac{h}{2}(\phi(q_n, p_{n-1/2}) + \phi(q_n, p_{n+1/2})) \tag{39}$$

By substituting the true solution (i.e., $q(t_{n+1})$ for q_{n+1}, etc.) in (38) and expanding in Taylor series, it is easy to see that (38)-(39) has a local error of $O(h^3)$. Standard arguments yield second order global convergence (see, e.g., Hairer, Nørsett and Wanner [14], Theorem 3.6).

Proof of Theorem 5.1:

Expand g around $q = q_n$ and evaluate at $q = q_{n+1}$ to get

$$g(q_{n+1}) = g(q_n) + g'_n(q_{n+1} - q_n) + \frac{1}{2}g''[q_{n+1} - q_n, q_{n+1} - q_n] \tag{40}$$

where we have written g'_n for $g'(q_n)$.

Making use of the equations definining **VS** we may write:

$$g'_n p_{n+1/2} = -\frac{h}{2}g''[p_{n+1/2}, p_{n+1/2}]$$

Next, introducing $p_{n+1/2}$ from (13), we get an equation that can be solved for λ_n.

$$\begin{aligned} \lambda_n &= \frac{1}{h^2}(g'_n g'^t_n)^{-1}\Big[-hg'_n p_{n-1/2} \\ &\quad +h^2 g'_n \nabla_q V(q_n) - (h^2/2)g''[p_{n+1/2}, p_{n+1/2}]\Big] \end{aligned} \tag{41}$$

Reintroducing this expression for λ in (13), and simplifying somewhat, we arrive at equations

$$q_{n+1} = q_n + hp_{n+1/2} \tag{42}$$

$$p_{n+1/2} = (I - \mathcal{H}_n)p_{n-1/2} - h(I - \mathcal{H}_n)\nabla_q V(q_n) \tag{43}$$

$$-\frac{h}{2}g'^t_n(g'_n g'^t_n)^{-1}g''[p_{n+1/2}, p_{n+1/2}] \tag{44}$$

Now evaluate the second-order expansion of g at q_{n-1} and use (12) at the previous step to get

$$g'_n p_{n-1/2} = \frac{h}{2}g''[p_{n-1/2}, p_{n-1/2}]$$

This leads, finally, to

$$q_{n+1} = q_n + hp_{n+1/2} \tag{45}$$

$$\begin{aligned} p_{n+1/2} &= p_{n-1/2} - h(I - \mathcal{H}_n)\nabla_q V(q_n) \\ &\quad -\frac{h}{2} g_n'^t (g_n' g_n'^t)^{-1} \Big(g''[p_{n+1/2}, p_{n+1/2}] + g''[p_{n-1/2}, p_{n-1/2}] \Big) \end{aligned} \tag{46}$$

We recognize (45)-(46) as nothing other than the discretization (38)-(39) applied to the underlying ODE (33)-(34). □