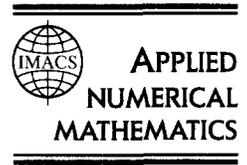




ELSEVIER

Applied Numerical Mathematics 29 (1999) 19–30



# Symplectic variable step size integration for $N$ -body problems

David J. Hardy<sup>a,1,2</sup>, Daniel I. Okunbor<sup>b,\*,1</sup>, Robert D. Skeel<sup>a,2</sup>

<sup>a</sup> *Department of Computer Science and Beckman Institute, University of Illinois at Urbana-Champaign,  
1304 West Springfield Avenue, Urbana, IL 61801-2987, USA*

<sup>b</sup> *Department of Computer Science, University of Missouri-Rolla, 1870 Miner Circle, Rolla, MO 65401-0350, USA*

---

## Abstract

Multiple time stepping can be applied to the leapfrog/Störmer/Verlet integrator so as to effect a variable step size algorithm. The strategy maintains the symplecticness, time-reversibility, and second-order accuracy of the leapfrog method. This method can be applied to 2-body central force interactions by partitioning them into distance classes and smoothly decomposing the potential energy into the sum of potential functions for the respective classes. The algorithm described here is much more efficient than leapfrog with very small step sizes and more accurate than leapfrog with larger step sizes. © 1999 Elsevier Science B.V. and IMACS. All rights reserved.

---

## 1. Introduction

The numerical time integration of dynamical Hamiltonian systems of ordinary differential equations (ODEs) is of considerable importance in areas of scientific research such as molecular dynamics and astrophysics. The goal is to run long time integrations of large systems in order to gain a better understanding of physical properties. Symplectic integration methods have been generally accepted as superior to non-symplectic methods for the integration of Hamiltonian systems [16], because symplectic integrators preserve properties that are intrinsic to Hamiltonian systems, the loss of which is extremely detrimental to long time integrations.

We are interested in efficient symplectic integration methods for solving the  $N$ -body problem in molecular dynamics. The preferred method has been the leapfrog/Störmer/Verlet method [22] which is known to be symplectic, time-reversible, and second-order accurate. The leapfrog method has the advantages of being both explicit and requiring only a single force evaluation per time step. Computing the force is the most expensive part of the integration. Considering the electrostatic force interactions between all pairs of atoms, a direct force evaluation requires computing  $N(N - 1)/2 = O(N^2)$  interactions which becomes prohibitively expensive for large  $N$ . The proposed modifications to the

---

\* Corresponding author.

<sup>1</sup> Supported in part by NSF Grant CCR-9408973.

<sup>2</sup> Supported in part by NSF Grant DMS-9600088, NSF Grant BIR-9318159, and NIH Grant P41RR05969.

leapfrog method greatly increase the efficiency of the integrator when directly computing the electrostatic forces and they can be combined with techniques such as the fast multipole algorithm (FMA) [4].

There has been much interest in the development of symplectic variable step size methods. Analysis in [18] shows that a fixed step size symplectic integrator will lose its symplecticness, in the traditional sense, when used with a variable step size. This is consistent with the experimental results of Calvo and Sanz-Serna [2], although results from Okunbor [14] suggest otherwise for second-order integrators. McLachlan and Scovel [13] present developing efficient symplectic variable step size methods as an open problem.

One approach is to use multiple time stepping techniques [19]. The potential energy function is decomposed with different parts integrated using different step sizes so that the overall method defines a symplectic transformation. Multiple time stepping was applied to the leapfrog method by Grubmüller et al. [5] in order to increase its efficiency. Tuckerman et al. [21] independently proposed the same method for its reversibility properties and introduced smooth force splittings. Skeel and Biesiadecki [17] first recognized that this method was in fact symplectic and effectively produces variable step size integration. Lee et al. [11] discussed this method for astrophysics applications and suggested different force decompositions. There are other variable step size modifications to the leapfrog method that are time-reversible but not symplectic [9,10].

Hairer [6] has described a different approach to symplectic variable step size integration, discovered independently by Reich [15], that works for all one-step methods. The Hamiltonian is transformed so that applying a constant step size to the new Hamiltonian solves the original system with variable step sizes. When this method is applied to separable Hamiltonians, the transformed Hamiltonian is no longer separable and instead requires implicit solution techniques. Unlike the method in [17], Hairer's method does not use multiple time stepping and seems to be of limited utility to applications such as molecular dynamics, in which the wide range of time scales present makes it more natural to integrate different interactions using different step sizes.

In this paper, we present the algorithmic details for the multiple time stepping approach. The straightforward implementation of the algorithm as described in [17] fails to achieve the efficiencies expected of a variable step size algorithm. Here, we present two devices for achieving the expected efficiencies:

- (1) Tables are utilized to evaluate a weighted sum of partial forces for a single interaction at a cost comparable to the direct evaluation of that interaction so that using force partitioning does not incur additional cost.
- (2) Particle speeds are monitored to avoid most redundant distance calculations associated with partial forces that are zero.

Consider the special separable Hamiltonian for an  $N$ -particle system in 3D

$$H(\mathbf{q}, \mathbf{p}) = \frac{1}{2} \mathbf{p}^T \mathbf{M}^{-1} \mathbf{p} + V(\mathbf{q}), \quad (1)$$

where

$$V(\mathbf{q}) = \sum_{i < j} V_{ij}(\|\mathbf{q}_j - \mathbf{q}_i\|) \quad (2)$$

is the total potential energy,  $\mathbf{M}$  is the diagonal mass matrix (with each mass replicated 3 times), and  $\mathbf{q}, \mathbf{p} \in \mathbb{R}^{3N}$  are respectively the positions and momenta with  $3N$  degrees of freedom. We use  $\mathbf{q}_i, \mathbf{p}_i$  and

$m_i$  to refer to the position, momentum, and mass of the  $i$ th particle, respectively. The first term on the right of (1) is the kinetic energy. The system of first-order ODEs associated with (1) is

$$\dot{\mathbf{q}} = M^{-1} \mathbf{p}, \quad \dot{\mathbf{p}} = \mathbf{f}(\mathbf{q}), \quad (3)$$

where the force is  $\mathbf{f}(\mathbf{q}) = -\nabla V(\mathbf{q})$ .

The leapfrog/Störmer/Verlet method [22] integrates the system (3) from time  $nh$  to time  $(n+1)h$ . Starting the step with values  $\mathbf{q}^n$ ,  $\mathbf{p}^n$  and  $\mathbf{f}^n = \mathbf{f}(\mathbf{q}^n)$  obtained from the previous step, we compute

$$\mathbf{p}^{n+1/2} = \mathbf{p}^n + \frac{1}{2}h\mathbf{f}^n, \quad (4)$$

$$\mathbf{q}^{n+1} = \mathbf{q}^n + hM^{-1}\mathbf{p}^{n+1/2}, \quad (5)$$

$$\mathbf{f}^{n+1} = \mathbf{f}(\mathbf{q}^{n+1}), \quad (6)$$

$$\mathbf{p}^{n+1} = \mathbf{p}^{n+1/2} + \frac{1}{2}h\mathbf{f}^{n+1}. \quad (7)$$

For a detailed discussion of the conservation properties of the leapfrog method, see [1,20].

## 2. The variable step size leapfrog method

Multiple time steps can be applied in a manner that effectively gives a variable step size method [17], preserving the favorable properties of the leapfrog method while reducing the amount of force computation.

### 2.1. Definition

We begin by choosing a largest fundamental step size  $h$  and a sequence of integer multiples  $\pi_k$  for  $k = 0, 1, \dots, \ell_{\max}$  such that  $\pi_0 = 1$  and  $\pi_{k+1}/\pi_k$  is an integer greater than 1. An efficient implementation might use  $\pi_k = 2^k$ . The value  $\ell_{\max}$  may be chosen arbitrarily large but fixed. The potential energy is partitioned into the sum

$$V(\mathbf{q}) = \sum_{k=0}^{\ell_{\max}} V^{[k]}(\mathbf{q}), \quad (8)$$

with corresponding partial forces

$$\mathbf{f}^{[k]}(\mathbf{q}) = -\nabla V^{[k]}(\mathbf{q}). \quad (9)$$

The partitioning is chosen so that the partial forces are stronger (shorter range) as  $k$  increases, with each  $\mathbf{f}^{[k]}$  to be used with step size  $h/\pi_k$ .

Step numbering is used here to count the number of fundamental steps of size  $h$  taken. For a given position vector  $\mathbf{q}$  in phase space, the value  $\ell$  is taken to be large enough so that the force terms  $\mathbf{f}^{[\ell+1]} = \mathbf{f}^{[\ell+2]} = \dots = \mathbf{f}^{[\ell_{\max}]} = 0$ , and  $h/\pi_\ell$  is taken to be the effective *micro-step size* for this *micro-step* of the integration. The step number can then be expressed as  $\sigma = n + m/\pi_\ell$  for some integer  $0 \leq m < \pi_\ell$ . The force to be computed is a weighted sum of partial force functions numbered  $0, 1, \dots, \ell$ ,

$$\mathbf{f}(\mathbf{q}; \sigma, \ell) = \sum_{k=0}^{\ell} w_k(\sigma) \mathbf{f}^{[k]}(\mathbf{q}), \quad (10)$$

where

$$w_k(\sigma) = \begin{cases} \frac{1}{\pi_k}, & \text{if } \pi_k \sigma \text{ is an integer,} \\ 0, & \text{otherwise.} \end{cases} \quad (11)$$

Intuitively, the method uses the leapfrog technique to integrate the force  $f^{[k]}$  with an effective step size  $h/\pi_k$ , hence the weights given by (11). A single micro-step of this method, expressed algorithmically to avoid complicated superscripting, is analogous to the leapfrog method:

$$\mathbf{p} \leftarrow \mathbf{p} + \frac{1}{2}h\mathbf{f}, \quad (12)$$

$$\mathbf{q} \leftarrow \mathbf{q} + \frac{h}{\pi_\ell}M^{-1}\mathbf{p}, \quad (13)$$

$$\sigma \leftarrow \sigma + \frac{1}{\pi_\ell}, \quad (14)$$

$$\mathbf{f} \leftarrow \mathbf{f}(\mathbf{q}; \sigma, \ell), \quad (15)$$

$$\mathbf{p} \leftarrow \mathbf{p} + \frac{1}{2}h\mathbf{f}. \quad (16)$$

The method is variable step size with micro-step size  $h/\pi_\ell$ , allowing  $\ell$  to be adjusted during the force computation. The value of  $\ell$  can be no smaller than the smallest  $k$  such that  $\pi_k\sigma$  is an integer and must remain large enough so that the highest numbered nonzero partial force is evaluated on the next micro-step. The integration is symplectic since the overall transformation is equivalent to a multiple time stepping scheme with smallest step size  $h/\pi_{\ell_{\max}}$  [7,8,17]. This scheme is second-order accurate and time-reversible [7,8,21], and these properties are maintained if  $\ell$  is adjusted appropriately (see Section 2.3).

## 2.2. Decomposing the potential energy function

Maintaining symplecticness with multiple time step leapfrog is achieved by splitting the potential energy function. The potential energy function (2) for any 2-body central force can be decomposed by partitioning the interactions into *distance classes*. We demonstrate this idea for the inverse square law, in which the  $ij$ th potential energy term is

$$V_{ij}(\mathbf{q}) = -\frac{c_{ij}}{r_{ij}}, \quad (17)$$

where  $r_{ij} = \|\mathbf{q}_j - \mathbf{q}_i\|$  and  $c_{ij}$  is an appropriate constant. First, we choose a decreasing sequence of positive cutoff radii

$$r_1 > r_2 > \cdots > r_{\ell_{\max}}, \quad (18)$$

where it is convenient to define  $r_0 = +\infty$  and  $r_{\ell_{\max}+1} = 0$ . One useful approach is to describe the cutoff radii by a geometric progression in which we choose the outer radius  $r_1$ , and the progression ratio  $0 < \rho < 1$ , and define  $r_k = \rho^{k-1}r_1$  for  $k = 2, \dots, \ell_{\max}$ . The  $ij$ th interaction is assigned to class  $k$  if  $r_{k+1} \leq r_{ij} < r_k$ . Notice that this approach places the stronger force interactions into higher numbered classes so that they are evaluated more often.

We create an artificial splitting of each potential energy term. For simplicity, let  $r = r_{ij}$  and  $c = c_{ij}$  and define a sequence of splitting functions

$$\phi_k(r) = \begin{cases} T_k(r), & \text{if } 0 < r < r_k, \\ V(r), & \text{if } r_k \leq r, \end{cases} \quad (19)$$

for  $k = 1, 2, \dots, \ell_{\max}$ , where  $T_k(r)$  is a smoothing of  $V(r)$  for small  $r$ . To ensure that the functions  $T_k(\|\mathbf{q}_j - \mathbf{q}_i\|)$  have bounded second partial derivatives [17], we use the first two terms of the Taylor series expansion of  $V(\sqrt{s})$  about  $s = r_k^2$ , yielding

$$T_k(r) = -\frac{c}{r_k} + \frac{c}{2r_k^3}(r^2 - r_k^2) = -\frac{c}{r_k^3}\left(\frac{3}{2}r_k^2 - \frac{1}{2}r^2\right). \tag{20}$$

We define the potential class terms from the splitting functions by

$$V^{[0]}(r) = \phi_1(r) = \begin{cases} -\frac{c}{r_1^3}\left(\frac{3}{2}r_1^2 - \frac{1}{2}r^2\right), & \text{if } 0 < r < r_1, \\ -\frac{c}{r}, & \text{if } r_1 \leq r, \end{cases} \tag{21}$$

$$V^{[k]}(r) = \phi_{k+1}(r) - \phi_k(r) = \begin{cases} \frac{c}{r_k^3}\left(\frac{3}{2}r_k^2 - \frac{1}{2}r^2\right) - \frac{c}{r_{k+1}^3}\left(\frac{3}{2}r_{k+1}^2 - \frac{1}{2}r^2\right), & \text{if } 0 < r < r_{k+1}, \\ \frac{c}{r_k^3}\left(\frac{3}{2}r_k^2 - \frac{1}{2}r^2\right) - \frac{c}{r}, & \text{if } r_{k+1} \leq r < r_k, \\ 0, & \text{if } r_k \leq r, \end{cases}$$

for  $k = 1, 2, \dots, \ell_{\max} - 1$ , (22)

$$V^{[\ell_{\max}]}(r) = V(r) - \phi_{\ell_{\max}}(r) = \begin{cases} \frac{c}{r_{\ell_{\max}}^3}\left(\frac{3}{2}r_{\ell_{\max}}^2 - \frac{1}{2}r^2\right) - \frac{c}{r}, & \text{if } 0 < r < r_{\ell_{\max}}, \\ 0, & \text{if } r_{\ell_{\max}} \leq r, \end{cases} \tag{23}$$

so that each class term is piecewise defined over its respective class interval with  $C^1$  smoothness at the join points. Other choices of splitting functions are discussed in [7,8,11,17].

The potential energy function for class  $k$  is the sum of class terms taken over all interactions

$$V^{[k]}(\mathbf{q}) = \sum_{i < j} V_{ij}^{[k]}(r_{ij}). \tag{24}$$

The  $i$ th force component for class  $k$  is the sum of the pairwise forces on particle  $i$  ( $\mathbf{f}_{ij}^{[k]}(\mathbf{q})$  is the force on particle  $i$  due to particle  $j$ ):

$$\mathbf{f}_i^{[k]}(\mathbf{q}) = \sum_{j \neq i} \mathbf{f}_{ij}^{[k]}(\mathbf{q}), \tag{25}$$

$$\mathbf{f}_{ij}^{[0]}(\mathbf{q}) = -\nabla_{\mathbf{q}_i} V_{ij}^{[0]}(r_{ij}) = \begin{cases} c_{ij} \left( \frac{1}{r_1^3} \right) (\mathbf{q}_j - \mathbf{q}_i), & \text{if } 0 < r_{ij} < r_1, \\ c_{ij} \left( \frac{1}{r_{ij}^3} \right) (\mathbf{q}_j - \mathbf{q}_i), & \text{if } r_1 \leq r_{ij}, \end{cases} \quad (26)$$

$$\mathbf{f}_{ij}^{[k]}(\mathbf{q}) = -\nabla_{\mathbf{q}_i} V_{ij}^{[k]}(r_{ij}) = \begin{cases} c_{ij} \left( \frac{1}{r_{k+1}^3} - \frac{1}{r_k^3} \right) (\mathbf{q}_j - \mathbf{q}_i), & \text{if } 0 < r_{ij} < r_{k+1}, \\ c_{ij} \left( \frac{1}{r_{ij}^3} - \frac{1}{r_k^3} \right) (\mathbf{q}_j - \mathbf{q}_i), & \text{if } r_{k+1} \leq r_{ij} < r_k, \\ 0, & \text{if } r_k \leq r_{ij}, \end{cases}$$

for  $k = 1, 2, \dots, \ell_{\max} - 1$ , (27)

$$\mathbf{f}_{ij}^{[\ell_{\max}]}(\mathbf{q}) = -\nabla_{\mathbf{q}_i} V_{ij}^{[\ell_{\max}]}(r_{ij}) = \begin{cases} c_{ij} \left( \frac{1}{r_{ij}^3} - \frac{1}{r_{\ell_{\max}}^3} \right) (\mathbf{q}_j - \mathbf{q}_i), & \text{if } 0 < r_{ij} < r_{\ell_{\max}}, \\ 0, & \text{if } r_{\ell_{\max}} \leq r_{ij}. \end{cases} \quad (28)$$

The test for class membership of an interaction is best done using the square of the distance

$$r_{k+1}^2 \leq r_{ij}^2 < r_k^2, \quad (29)$$

which avoids a square root operation which is potentially costly for  $N$  large.

### 2.3. Evaluating the force function

A single force interaction can be evaluated in a time which is independent of the number of classes. The  $ij$ th force is expressed as a sum

$$\mathbf{f}_{ij}(\mathbf{q}; \sigma, \ell) = \sum_{k=0}^{\ell} w_k(\sigma) \mathbf{f}_{ij}^{[k]}(\mathbf{q}). \quad (30)$$

As seen from (26)–(28), the constants and direction vectors can be factored from the summation. Note,  $r_{ij}$  is contained in exactly one of the disjoint subintervals, providing the only remaining term in the summation with a non-constant value as well as the only square root that needs to be evaluated for this force interaction. This implies all possible sums of constant terms should be precomputed for efficient evaluation.

As an example, suppose the interaction is in class  $k$ , that is,  $r_{k+1} \leq r_{ij} < r_k$ , and let  $\gamma$  be the lowest numbered class for which  $\pi_\gamma \sigma$  is an integer. Assume that  $0 < \gamma < k$ . Evaluating  $\mathbf{f}_{ij}$  gives

$$\begin{aligned} \mathbf{f}_{ij}(\mathbf{q}; \sigma, \ell) = c_{ij} (\mathbf{q}_j - \mathbf{q}_i) & \left[ 0 \cdot \frac{1}{r_1^3} + 0 \cdot \left( \frac{1}{r_2^3} - \frac{1}{r_1^3} \right) + \dots + 0 \cdot \left( \frac{1}{r_\gamma^3} - \frac{1}{r_{\gamma-1}^3} \right) \right. \\ & \left. + \frac{1}{\pi_\gamma} \left( \frac{1}{r_{\gamma+1}^3} - \frac{1}{r_\gamma^3} \right) + \dots + \frac{1}{\pi_{k-1}} \left( \frac{1}{r_k^3} - \frac{1}{r_{k-1}^3} \right) + \frac{1}{\pi_k} \left( \frac{1}{r_{ij}^3} - \frac{1}{r_k^3} \right) \right] \end{aligned}$$

$$\begin{aligned}
 &= c_{ij}(\mathbf{q}_j - \mathbf{q}_i) \left[ \frac{1}{\pi_k r_{ij}^3} + \sum_{\eta=\gamma}^{k-1} \frac{1}{r_{\eta+1}^3} \left( \frac{1}{\pi_\eta} - \frac{1}{\pi_{\eta+1}} \right) - \frac{1}{\pi_\gamma r_\gamma^3} \right] \\
 &= c_{ij}(\mathbf{q}_j - \mathbf{q}_i) \left( \frac{1}{\pi_k r_{ij}^3} + \alpha_{k\gamma} \right).
 \end{aligned} \tag{31}$$

Using this approach, all possible additive constants are given by

$$\alpha_{k\gamma} = \begin{cases} 0, & \text{if } 0 = \gamma = k, \\ \sum_{\eta=0}^{k-1} \frac{1}{r_{\eta+1}^3} \left( \frac{1}{\pi_\eta} - \frac{1}{\pi_{\eta+1}} \right), & \text{if } 0 = \gamma < k, \\ \sum_{\eta=\gamma}^{k-1} \frac{1}{r_{\eta+1}^3} \left( \frac{1}{\pi_\eta} - \frac{1}{\pi_{\eta+1}} \right) - \frac{1}{\pi_\gamma r_\gamma^3}, & \text{if } 0 < \gamma < k, \\ -\frac{1}{\pi_\gamma r_\gamma^3}, & \text{if } 0 < \gamma = k, \end{cases} \tag{32}$$

for each  $k = 0, 1, \dots, \ell_{\max}$ . In practice,  $\ell_{\max}$  will be small enough (perhaps  $\ell_{\max} < 20$ ) so that the space required by the  $\alpha_{k\gamma}$  table is insignificant.

We need to limit the amount of interaction class testing. In order to keep track of the interactions, each class  $k$  is provided with a list  $L_k$  of interaction pairs. We assume that an interaction may move only to a consecutive class between force evaluations, that is, an interaction may not skip over classes. This assumption allows us to integrate with a micro-step size  $h/\pi_{\ell+1}$  based on the highest numbered nonempty class  $\ell$ , eliminating most of the redundant computation associated with monitoring interaction pairs for the partial forces from  $\mathbf{f}(\mathbf{q}; \sigma, \ell)$  that are zero. Given the smallest  $\gamma > 0$  such that  $\pi_\gamma \sigma$  is an integer and the highest numbered nonempty class  $\ell$ , to implement the assumption correctly we must check class  $\gamma - 1$  for possible interaction transitions and must also monitor transitions into class  $\ell + 1$  using a step size of  $h/\pi_{\ell+1}$ . The position update (13) and the step number update (14) must be modified to accommodate this smaller step size. Note, violation of this assumption could ruin the symplecticness of the integrator, since the forces contributed by the different classes would be evaluated incorrectly. (If an interaction does skip over a class, probably the cutoff radii are too close together.)

We can validate the assumption on interaction class transitions by monitoring the speed of each particle. For an interaction to skip a class  $k$ , the distance between the two particles must change by at least  $r_k - r_{k+1}$  in time  $h/\pi_k$ . This means that the assumption holds if the speed of every particle  $i$  is bounded,

$$\frac{\|\mathbf{p}_i\|}{m_i} \leq \frac{1}{2} \left( \frac{r_k - r_{k+1}}{h/\pi_k} \right), \tag{33}$$

for each  $k = 1, \dots, \ell_{\max}$ . The smallest such bound can be checked in linear time immediately following the momentum update (12).

If the speed check (33) ever fails, then we can save symplecticness by using the particle velocities to predict the next appropriate micro-step size. Suppose the check fails for the  $i$ th particle. First, we seek the smallest  $s > 1$  that makes

$$\frac{\|\mathbf{p}_i\|}{m_i} \leq \frac{1}{2} \left( \frac{r_k - r_{k+s}}{h/\pi_k} \right) \tag{34}$$

for each  $k = 1, \dots, \ell_{\max} + 1 - s$ , or just choose  $s = \ell_{\max}$  if all these bounds are violated. The value  $s$  is the skip bound, indicating the number of classes away that an interaction involving particle  $i$  might transit after the next position update. The next position update (13) and step number update (14) should then be performed using the micro-step size  $h/\pi_{\ell+s}$  (or the smallest step size  $h/\pi_{\ell_{\max}}$  if  $\ell + s > \ell_{\max}$ ). The force computation checks up to  $s$  classes away for interaction transitions from the lower numbered classes.

Correct implementation of a single micro-step of the variable step size integration follows:

$$\mathbf{p} \leftarrow \mathbf{p} + \frac{1}{2}h\mathbf{f}, \quad (35)$$

$$s \leftarrow \min_i \left\{ a \left| \frac{\|\mathbf{p}_i\|}{m_i} \leq \frac{1}{2} \left( \frac{r_k - r_{k+a}}{h/\pi_k} \right) \text{ for } k = 0, \dots, \ell_{\max} + 1 - a \right. \right\}, \quad (36)$$

$$\mathbf{q} \leftarrow \mathbf{q} + \frac{h}{\pi_{\min\{\ell+s, \ell_{\max}\}}} M^{-1} \mathbf{p}, \quad (37)$$

$$\sigma \leftarrow \sigma + \frac{1}{\pi_{\min\{\ell+s, \ell_{\max}\}}}, \quad (38)$$

$$\mathbf{f} \leftarrow \mathbf{f}(\mathbf{q}; \sigma, \ell, s), \quad (39)$$

$$\mathbf{p} \leftarrow \mathbf{p} + \frac{1}{2}h\mathbf{f}. \quad (40)$$

Note,  $s$  is now a parameter to the force computation. The following algorithm details this computation.

**Input:**  $\mathbf{q} = (\mathbf{q}_1^T, \dots, \mathbf{q}_n^T)^T$  [position],  $\sigma$  [step number],  $\ell$  [highest class],  $s$  [skip bound]

**Output:**  $\mathbf{f} = (\mathbf{f}_1^T, \dots, \mathbf{f}_n^T)^T$  [force],  $\ell$  [updated highest class]

[ $+\infty = r_0 > r_1 > r_2 > \dots > r_{\ell_{\max}} > r_{\ell_{\max}+1} = 0$  are the cutoff radii]

[ $L_0, L_1, L_2, \dots, L_{\ell_{\max}}$  are the lists of interaction pairs  $(i, j)$ ]

[ $L'_0, L'_1, L'_2, \dots, L'_{\ell_{\max}}$  are initially empty lists]

$\gamma \leftarrow \min\{k \mid \pi_k \sigma \text{ is an integer}\}$

$\mathbf{f} \leftarrow 0$

$\zeta \leftarrow \max\{\gamma - s, 0\}$

**for**  $k \leftarrow \zeta, \zeta + 1, \dots, \gamma - 1$  **do**

[check upward transitions from lower classes]

**for each**  $(i, j) \in L_k$  **do**

remove  $(i, j)$  from  $L_k$

$\delta \leftarrow \|\mathbf{q}_j - \mathbf{q}_i\|^2$

[use the square of the distance]

**if**  $\delta < r_{k+1}^2$  **then**

[move to a higher class]

$a \leftarrow 1$

**while**  $\delta < r_{k+1+a}^2$  **do**

$a \leftarrow a + 1$

**end while**

**if**  $k + a \geq \gamma$  **then**

evaluate  $(i, j)$  in class  $k + a$

[update  $\mathbf{f}_i$  and  $\mathbf{f}_j$ ]

**end if**

insert  $(i, j)$  into  $L'_{k+a}$

[( $i, j$ ) is not processed again]

**else**

[keep in same class]

insert  $(i, j)$  into  $L'_k$

[( $i, j$ ) is not processed again]

```

    end if
  end for
end for
for  $k \leftarrow \gamma, \gamma + 1, \dots, \ell$  do
    [locate and evaluate higher class interactions]
  for each  $(i, j) \in L_k$  do
    remove  $(i, j)$  from  $L_k$ 
     $\delta \leftarrow \|q_j - q_i\|^2$ 
    [use the square of the distance]
    if  $\delta < r_{k+1}^2$  then
      [move to a higher class]
       $a \leftarrow 1$ 
      while  $\delta < r_{k+1+a}^2$  do
         $a \leftarrow a + 1$ 
      end while
      evaluate  $(i, j)$  in class  $k + a$ 
      [update  $f_i$  and  $f_j$ ]
      insert  $(i, j)$  into  $L'_{k+a}$ 
      [(i, j) is not processed again]
    else if  $\delta \geq r_k^2$  then
      [move to a lower class]
       $a \leftarrow 1$ 
      while  $\delta \geq r_{k-a}^2$  do
         $a \leftarrow a + 1$ 
      end while
      if  $k - a \geq \gamma$  then
        evaluate  $(i, j)$  in class  $k - a$ 
        [update  $f_i$  and  $f_j$ ]
      end if
      insert  $(i, j)$  into  $L'_{k-a}$ 
      [(i, j) is not processed again]
    else
      [keep in same class]
      evaluate  $(i, j)$  in class  $k$ 
      [update  $f_i$  and  $f_j$ ]
      insert  $(i, j)$  into  $L'_k$ 
      [(i, j) is not processed again]
    end if
  end for
end for
end for
 $\ell \leftarrow \max\{\max\{k \mid L'_k \neq \emptyset\}, \min\{k \mid \pi_k \sigma \text{ is an integer}\}\}$ 
[adjust  $\ell$ ]
for  $k \leftarrow \zeta, \zeta + 1, \dots, \ell$  do
   $L_k \leftarrow L'_k$ 
  [save the new interaction lists]
   $L'_k \leftarrow \emptyset$ 
  [reset the temporary lists]
end for

```

For efficiency in the first outer **for** loop of the algorithm, interactions in lower classes are moved only upward since they might affect the force computation. This means that a lower numbered pair list might not accurately reflect the interactions contained in its respective class. However, this will not affect the correctness of the algorithm since an interaction that should be in a higher numbered class will be placed properly before its evaluation. Since the two outer **for** loops are very similar, they could be combined but at the cost of redundant comparisons, so we decided to present the algorithm in its expanded form. This algorithm is quite efficient for the cases that we most expect. Interactions will change classes infrequently, so, typically, we bypass all the **while** loops, and seldom skip over classes, so the first outer **for** loop is almost never repeated more than once. An efficient implementation might decide to simply

use (33) to bound the particle speeds, with a warning message whenever the check fails, and modify the force computation so that an interaction is never moved by more than one class, which eliminates the **while** loops. In the worst case, whenever some particle's speed remains unbounded in the check (33), the algorithm degenerates to "pure" multiple time stepping with step size  $h/\pi_{\ell_{\max}}$  in which every interaction has to be checked for possible evaluation.

#### 2.4. Possible improvements

Choosing the cutoff radii as a geometric progression with ratio  $\rho$  allows some simplification. The difference between cutoff radii is now

$$r_k - r_{k+1} = \left(\frac{1}{\rho} - 1\right) \rho^k r_1. \quad (41)$$

Since  $\pi_k \geq 2^k$ , for (33) it is enough to bound the speed of each particle  $i$  by

$$\frac{\|\mathbf{p}_i\|}{m_i} \leq \left(\frac{1}{\rho} - 1\right) \frac{r_1}{2h} (2\rho)^k. \quad (42)$$

For  $\frac{1}{2} \leq \rho < 1$ , the bound is minimized at  $k = 0$ , so it would suffice to check

$$\frac{\|\mathbf{p}_i\|}{m_i} \leq \left(\frac{1}{\rho} - 1\right) \frac{r_1}{2h}. \quad (43)$$

In determining a suitable  $\rho$ , we can apply the rule suggested in [12] for linear problems, that the optimal step size is inversely proportional to the square root of the spectral radius of the Hessian of the potential function. Applying this idea to a single electrostatic interaction, the spectral radius of the Hessian of  $\|\mathbf{q}_j - \mathbf{q}_i\|^{-1}$  is  $4r_{ij}^{-3}$ , so we want to vary the step size  $\Delta t$  so that

$$(\Delta t)^2 \frac{4}{r_{ij}^3} = \text{constant}. \quad (44)$$

Assuming weights  $w_k(\sigma) = 1/\pi_k = 2^{-k}$ , we want to choose the cutoff radius  $r_k$  such that

$$\left(\frac{h}{2^{k-1}}\right)^2 \frac{4}{r_k^3} = \text{constant}, \quad (45)$$

which implies that

$$r_k = \text{constant} \cdot (2^{-2/3})^{k-1} = (2^{-2/3})^{k-1} r_1, \quad (46)$$

suggesting that the ratio  $\rho = 2^{-2/3}$  is optimal.

Assuming an upper bound on the density of particles in the system as well as a bound on their speeds, the cost of computing the short range interactions is proportional to  $Nr^3$  for a fixed cutoff radius  $r$ . For appropriate choices of class weights and cutoff radii, we can show that linear work per fundamental step  $h$  is done in evaluating the short range interactions, independent of the number of classes used. For instance, choosing weights  $1/\pi_k = 2^{-k}$  and cutoff radius ratio  $\rho = 2^{-2/3}$ , the amount of short range force computation performed in one step will be proportional to

$$\lim_{\ell_{\max} \rightarrow \infty} \sum_{k=1}^{\ell_{\max}} \pi_k N r_k^3 = 4N r_1^3 = O(N). \quad (47)$$

This idea can be extended to develop an efficient linear time and space algorithm for large systems. Define an outer bounding distance  $r_0 = r_1/\rho$  to use in monitoring long range interaction transitions from class 0. At the end of every fundamental step, we can use linear time algorithms to evaluate all the force interactions and reformulate the interaction lists within the bounding distance, for instance, FMA to compute the force and geometric hashing [3] to reformulate the interaction lists. It would no longer be necessary to store or check interactions that lie outside of  $r_0$  during the micro-steps, so the time and space used remain linear. A much greater efficiency is achieved over using standard leapfrog with a small step size and FMA force computation, because the time complexity constant for FMA is very large [4].

## References

- [1] S.P. Auerbach and A. Friedman, Long-time behavior of numerically computed orbits: small and intermediate timestep analysis of one-dimensional systems, *J. Comput. Phys.* 93 (1991) 189–223.
- [2] M.P. Calvo and J.M. Sanz-Serna, The development of variable-step symplectic integrators, with applications to the two-body problem, *SIAM J. Sci. Statist. Comput.* 14 (1993) 936–952.
- [3] G.C. Fox, M.A. Johnson, G.A. Lyzenga, S.W. Otto, J.K. Salmon and D.W. Walker, *Solving Problems on Concurrent Processors*, Vol. I: *General Techniques and Regular Problems* (Prentice-Hall, Englewood Cliffs, NJ, 1988).
- [4] L. Greengard, *The Rapid Evaluation of Potential Fields in Particle Systems* (MIT Press, Cambridge, MA, 1988).
- [5] H. Grubmüller, H. Heller, A. Windemuth and K. Schulten, Generalized Verlet algorithm for efficient molecular dynamics simulations with long-range interactions, *Molec. Sim.* 6 (1991) 121–142.
- [6] E. Hairer, Variable time step integration with symplectic methods, *Ann. Numer. Math.* 1 (1994) 102–132.
- [7] D.J. Hardy, Multiple time step symplectic integration, Master's Thesis, University of Missouri-Rolla (1997).
- [8] D.J. Hardy and D.I. Okunbor, Symplectic multiple time step integration, *J. Comput. Phys.*, submitted.
- [9] W. Huang and B. Leimkuhler, The adaptive Verlet method, *SIAM J. Sci. Comput.* 18 (1997) 239–256.
- [10] P. Hut, J. Makino and S. McMillan, Building a better leapfrog, *Astrophys. J.* 443 (1995) L93–L96.
- [11] M.H. Lee, M.J. Duncan and H.F. Levison, Variable timestep integrators for long-term orbital integrations, in: *Computational Astrophysics, Proc. 12th Kingston Meeting* (1997).
- [12] T.R. Littell, R.D. Skeel and M. Zhang, Error analysis of symplectic multiple time stepping, *SIAM J. Numer. Anal.* 34 (1997) 1762–1807.
- [13] R.I. McLachlan and C. Scovel, A survey of open problems in symplectic integration, in: J.E. Marsden, G.W. Patrick and W.F. Shadwick, eds., *Integration Algorithms and Classical Mechanics* (American Mathematical Society, Providence, RI, 1996).
- [14] D.I. Okunbor, Variable step size does not harm second-order integrators for Hamiltonian systems, *J. Comput. Appl. Math.* 47 (1993) 273–275.
- [15] S. Reich, Backward error analysis for numerical integrators, Konrad-Zuse-Zentrum für Informationstechnik Berlin, Preprint SC 96-21 (1996).
- [16] J.M. Sanz-Serna and M.P. Calvo, *Numerical Hamiltonian Problems* (Chapman and Hall, London, 1994).
- [17] R.D. Skeel and J.J. Biesiadecki, Symplectic integration with variable stepsize, *Ann. Numer. Math.* 1 (1994) 191–198.
- [18] R.D. Skeel and C.W. Gear, Does variable step size ruin a symplectic integrator?, *Physica D* 60 (1992) 311–313.
- [19] W.B. Streett, D.J. Tildesley and G. Saville, Multiple time-step methods in molecular dynamics, *Molec. Phys.* 35 (1978) 639–648.
- [20] S. Toxvaerd, Hamiltonians for discrete dynamics, *Phys. Rev.* 50 (1994) 2271–2274.

- [21] M. Tuckerman, B.J. Berne and G.J. Martyna, Reversible multiple time scale molecular dynamics, *J. Chem. Phys.* 97 (1992) 1990–2001.
- [22] L. Verlet, Computer ‘experiments’ on classical fluids I. Thermodynamical properties of Lennard–Jones molecules, *Phys. Rev.* 159 (1967) 98–103.