A SYMPLECTIC INTEGRATOR FOR HILL'S EQUATIONS

THOMAS QUINN[1], RANDALL P. PERRINE[2], DEREK C. RICHARDSON[2], AND RORY BARNES[1]
[1] Department of Astronomy, University of Washington, Box 351580, Seattle, WA 98195, USA; trq@astro.washington.edu
[2] Department of Astronomy, University of Maryland, Computer & Space Sciences Building, Stadium Drive College Park, MD 20742, USA

ABSTRACT

Hill's equations are an approximation that is useful in a number of areas of astrophysics including planetary rings and planetesimal disks. We derive a symplectic method for integrating Hill's equations based on a generalized leapfrog. This method is implemented in the parallel *N*-body code, *PKDGRAV*, and tested on some simple orbits. The method demonstrates a lack of secular changes in orbital elements, making it a very useful technique for integrating Hill's equations over many dynamical times. Furthermore, the method allows for efficient collision searching using linear extrapolation of particle positions.

Key words: methods: miscellaneous – methods: numerical

1. INTRODUCTION

There are a number of situations in planetary dynamics that require the exploration of near-circular orbits. Current topics of interest in this category include planetary rings (Wisdom & Tremaine 1988) and planet formation (Tanga et al. 2004; Barnes et al. 2009, hereafter BQLR). In this circumstance, the equations of motion can be linearized about the circular orbit as was first done by Hill (1878) to study the lunar orbit. That is, the motions of bodies are described with respect to a Cartesian frame that is in uniform circular motion about a central body, and excursions from the center of the frame are small compared to the distance to the central body. In the absence of perturbations, the resulting Hill's equations describe simple epicyclic motion, and can also be used for disk dynamics (Goldreich & Lynden-Bell 1965; Julian & Toomre 1966) and the escape of stars from globular clusters (Heggie 2001).

A noticeable disadvantage of Hill's equations for numerical integration is that they contain a velocity-dependent force. Simulation codes for large *N*-body simulations, (e.g., Springel 2005; Wadsley et al. 2004) typically use the leapfrog integration scheme, which is second order, symplectic, and easy to implement. The leapfrog scheme can be modified to take velocity-dependent forces into account and still retains second order, as is done for Smoothed Particle Hydrodynamics (SPH); however, this destroys its symplectic nature.

The power of symplectic integrators is rooted in the property that any truncation error can be represented as a perturbing Hamiltonian. Hence for sufficiently small step size, the numerical system has conserved quantities similar to the integrals of motion of the physical system. That is, the numerical integration is an exact solution to an approximate Hamiltonian. This property is particularly important when following systems for many dynamical times such as the long-term evolution of the solar system, or investigating the stability of extrasolar planetary systems. In these situations, if the integrator introduces secular changes in the actions, the dynamics being investigated can be fundamentally changed. Hence, symplectic integrators are widely used in such investigations (Holman & Wisdom 1993; Levison & Duncan 1994; Malhotra 1995; Lee & Peale 2002; Rivera & Lissauer 2000). Both planetary rings and planetesimal dynamics are systems that evolve over large numbers of dynamical times, and therefore may also benefit from the use of symplectic integrators.

A symplectic integrator for Hill's equations was introduced by Heggie (2001) in the context of escape of stars from globular clusters. However, in that work, the integrator is not actually put to use; instead, the orbits were calculated using a Hermite integrator. The symplectic integrator was expressed as an implicit set of equations which, however, could be solved explicitly. As shown below, a canonical transformation can significantly simplify the Hamiltonian, and therefore simplify the resulting integrator.

Saha & Tremaine (1992) introduced a formalism (also see Wisdom & Holman 1991) for deriving symplectic integrators of a generalized leapfrog type by separating the Hamiltonian into parts that can be integrated exactly and then using commutator algebra to combine these solutions into a symplectic solution to the full problem. Quinn et al. (1997) showed how this technique could be used for cosmological simulations that involve a time-dependent Hamiltonian. This formalism has also been used to construct higher order integrators (Laskar & Robutel 2001; Chambers & Murison 2000), integrators that handle close encounters (Chambers 1999; Duncan et al. 1998), and integrators that efficiently integrate problems with a large dynamic range (Saha & Tremaine 1994; McNeil & Nelson 2009). Here, we will apply the technique to Hill's equations. In Section 2, we present the Hamiltonian formulation of Hill's equations from which, in Section 3, we derive a symplectic integrator suitable for use in a large *N*-body code. In Section 4, we describe its implementation in the *PKDGRAV N*-body code (Stadel 2001), explicitly stating the algorithm for timestepping a simulation; and in Section 5, we perform tests appropriate for the application of planetesimal dynamics in the early solar system. Section 6 contains a short discussion and summary.

2. HAMILTONIAN FORMULATION

The Lagrangian for Hill's equations in the orbital plane is

$$\mathcal{L} = \frac{1}{2}\left[(\dot{x} - \Omega y)^2 + (\dot{y} + \Omega x)^2\right] + \frac{1}{2}\Omega^2(2x^2 - y^2) - \Phi(x, y), \tag{1}$$

(Heggie 2001) where *x* and *y* are, respectively, the distances perpendicular to and along the direction of rotation from the center of a frame in circular motion with angular speed Ω. Φ is the potential due to other forces, e.g., interactions with other particles. In all that follows, we will neglect the motion in the *z*-direction since it is trivial to integrate in the standard way.

Lagrange's equations give the standard Hill's equations of motion,

$$\ddot{x} - 2\Omega\dot{y} - 3\Omega^2 x = -\frac{\partial\Phi}{\partial x}, \qquad (2)$$

$$\ddot{y} + 2\Omega\dot{x} = -\frac{\partial\Phi}{\partial y}. \qquad (3)$$

In this form, the presence of the velocity-dependent terms requires a modification to the leapfrog method such that a predicted velocity is used in the estimate of the final acceleration. This maintains second order, but is obviously not time reversible and destroys the symplectic nature of leapfrog. However, as we show below, a symplectic integrator can be derived for this system.

To construct a symplectic integrator, we first derive the Hamiltonian form of the equations of motion. From their definitions, the canonical momenta are

$$p_x \equiv \frac{\partial\mathcal{L}}{\partial\dot{x}} = \dot{x} - \Omega y, \qquad (4)$$

$$p_y \equiv \frac{\partial\mathcal{L}}{\partial\dot{y}} = \dot{y} + \Omega x, \qquad (5)$$

and the Hamiltonian is

$$H(x, y, p_x, p_y) = \frac{p_x^2}{2} + \frac{p_y^2}{2} + \Omega(yp_x - xp_y)$$
$$- \frac{1}{2}\Omega^2(2x^2 - y^2) + \Phi(x, y). \qquad (6)$$

Now consider a new set of canonical coordinates, (X, Y, P_x, P_y), derived from the generating function

$$S_2(x, y, P_x, P_y) = xP_x + yP_y - \Omega xy. \qquad (7)$$

The rules of canonical transformations then give

$$p_x = \frac{\partial S_2}{\partial x} = P_x - \Omega y, \quad p_y = \frac{\partial S_2}{\partial y} = P_y - \Omega x, \qquad (8)$$

and

$$X = \frac{\partial S_2}{\partial P_x} = x, \quad Y = \frac{\partial S_2}{\partial P_y} = y. \qquad (9)$$

In terms of the original positions and velocities, these new canonical coordinates are $x, y, P_x = \dot{x}$, and $P_y = \dot{y} + 2\Omega x$. The Hamiltonian in these coordinates is

$$H(x, y, P_x, P_y) = \frac{P_x^2}{2} + \frac{P_y^2}{2} - 2\Omega xP_y + \frac{\Omega^2 x^2}{2} + \Phi(x, y), \qquad (10)$$

a somewhat simpler form than Equation (6).

Hamilton's equations of motion are therefore

$$\dot{x} = P_x, \qquad (11)$$

$$\dot{y} = P_y - 2\Omega x, \qquad (12)$$

$$\dot{P}_x = 2\Omega P_y - \Omega^2 x - \frac{\partial\Phi}{\partial x}, \qquad (13)$$

$$\dot{P}_y = -\frac{\partial\Phi}{\partial y}. \qquad (14)$$

From these equations, it is obvious that P_y is constant in the absence of perturbing forces. This is equivalent to the conservation of angular momentum, and leads to the conserved quantity, $\sum_i P_{y_i}$, in a many-particle system. In particular, $\sum_i P_{y_i}$ is conserved in a collision between particles (Wisdom & Tremaine 1988), which will be useful when calculating collision outcomes (see below). However, for periodic boundary conditions, P_y will change as a particle crosses the boundary in x because of the shear across the box. Nevertheless, the time-averaged total P_y should be constant for a system that does not have a net motion in the x-direction (Wisdom & Tremaine 1988). Also from the equations of motion in this form, it is clear why Hill's equations are easy to integrate numerically. If the guiding center of the motion is at $x = 0$, then $P_y = 0$ for all time in the absence of perturbations, and the motion reduces to a harmonic oscillator with frequency Ω.

3. SYMPLECTIC INTEGRATORS

Normally, one can create a symplectic integrator by separating the Hamiltonian into exactly integral parts as in Saha & Tremaine (1992), but the presence of a velocity-dependent force makes this approach nontrivial in the case of Hill's problem. The Hamiltonian can be split as follows:

$$H = \underbrace{\frac{1}{2}\left(P_x^2 + P_y^2\right)}_{H_{\mathrm{FP}}} + \underbrace{-2\Omega xP_y + \frac{\Omega^2 x^2}{2} + \Phi(x, y)}_{H_{\mathrm{MDF}}}, \qquad (15)$$

where H_{FP} is the free particle Hamiltonian and H_{MDF} is considered to be the (momentum-dependent) "force" term. The H_{FP} Hamiltonian is easily solved and is just the motion of a particle with constant velocity:

$$\begin{aligned} x(t_0 + \tau) &= x(t_0) + \tau P_x(t_0), \\ y(t_0 + \tau) &= y(t_0) + \tau P_y(t_0), \\ P_x(t_0 + \tau) &= P_x(t_0), \\ P_y(t_0 + \tau) &= P_y(t_0), \end{aligned} \qquad (16)$$

where τ is the timestep and t_0 is the initial time. However, the equation of motion corresponding to the H_{MDF} Hamiltonian cannot be solved easily. This is because for this part of the Hamiltonian y is not constant ($\partial H_{\mathrm{MDF}}/\partial P_y \neq 0$), so one must evaluate the force along a trajectory determined by $\dot{y} = -2\Omega x$ to solve for P_y, and then use this to solve for P_x. This would prove intractable in a large simulation.

Instead, let us separate the Hamiltonian into a mixed term, H_{Mix}, and a momentum-independent force term, H_{MIF}, as follows:

$$H = \underbrace{\frac{1}{2}\left(P_x^2 + P_y^2\right) - 2\Omega xP_y}_{H_{\mathrm{Mix}}} + \underbrace{\frac{\Omega^2 x^2}{2} + \Phi(x, y)}_{H_{\mathrm{MIF}}}. \qquad (17)$$

H_{MIF} is easily integrated to give the equations of motion,

$$P_x(t_0 + \tau) = P_x(t_0) - \tau\left(\Omega^2 x(t_0) + \left.\frac{\partial\Phi}{\partial x}\right|_{t_0}\right),$$

$$P_y(t_0 + \tau) = P_y(t_0) - \tau\left.\frac{\partial\Phi}{\partial y}\right|_{t_0}. \qquad (18)$$

The mixed Hamiltonian, H_{Mix}, gives the equations of motion,

$$\dot{x} = P_x, \qquad (19)$$

$$\dot{y} = P_y - 2\Omega x, \tag{20}$$

$$\dot{P}_x = 2\Omega P_y, \tag{21}$$

$$\dot{P}_y = 0. \tag{22}$$

These can be integrated exactly as follows: P_y is a constant so $P_y(t) = P_y(t_0)$; $P_x(t)$ can now be solved; $x(t)$ can be solved once $P_x(t)$ is known; and finally $y(t)$ can be solved since we know $P_y(t)$ and $x(t)$. We therefore have

$$
\begin{aligned}
x(t_0 + \tau) &= x(t_0) + \tau P_x(t_0) + \tau^2 \Omega P_y(t_0), \\
y(t_0 + \tau) &= y(t_0) + \tau(P_y(t_0) - 2\Omega x(t_0)) \\
&\quad - \tau^2 \Omega P_x(t_0) - \tau^3 \frac{2}{3}\Omega^2 P_y(t_0), \\
P_x(t_0 + \tau) &= P_x(t_0) + \tau 2\Omega P_y(t_0), \\
P_y(t_0 + \tau) &= P_y(t_0).
\end{aligned}
\tag{23}
$$

These can be used to construct a second-order symplectic integrator exactly analogous to leapfrog by applying Equations (18) for half a timestep, Equations (23) for a full timestep, and Equations (18) for another half timestep. If we were simply integrating force equations this is straightforward to implement in a large N-body code. However, in the case of planetesimal and planetary ring dynamics, collisions between particles need to be detected. Current collision detection algorithms rely on the position updates being linear in time (Richardson et al. 2000), and certainly not cubic in time as in the above.

In an attempt to simplify the mixed equations of motion, let us separate H_{Mix} even further into the free particle Hamiltonian, H_{FP}, and the cross term, $H_C = -2\Omega x P_y$. That is,

$$H = \underbrace{\frac{1}{2}\left(P_x^2 + P_y^2\right)}_{H_{\text{FP}}} + \underbrace{-2\Omega x P_y}_{H_C} + \underbrace{\frac{\Omega^2 x^2}{2} + \Phi(x, y)}_{H_{\text{MIF}}}. \tag{24}$$

The cross Hamiltonian is easily integrable giving

$$x(t_0 + \tau) = x(t_0), \tag{25}$$

$$y(t_0 + \tau) = y(t_0) - \tau 2\Omega x(t_0), \tag{26}$$

$$P_x(t_0 + \tau) = P_x(t_0) + \tau 2\Omega P_y(t_0), \tag{27}$$

$$P_y(t_0 + \tau) = P_y(t_0). \tag{28}$$

Therefore, a first-order symplectic scheme presents itself as follows: (1) update the momenta using Equations (18); (2) update P_x using Equation (27) and the P_y from step 1; (3) update the y positions of the particles according to Equation (26). If collisions are being considered, they are searched for in this step; and (4) perform a standard position update using the free particle Hamiltonian (Equation (16)), again searching for collisions. In our implementation, steps 3 and 4 are combined into a single position update that includes the collision search.

The construction of a second-order scheme follows using the formalism of Saha & Tremaine (1992). If we refer to the evolution of phase space for a time τ under the Hamiltonians H_{MIF}, H_{FP}, and H_C as $\text{MIF}(\tau)$ (Equation (18)), $\text{FP}(\tau)$ (Equation (16)), and $C(\tau)$ (Equations (25)–(28)), respectively, then the combination of operators $\text{MIF}(\tau/2)C(\tau/2)\text{FP}(\tau)C(\tau/2)\text{MIF}(\tau/2)$ will evolve the system for a timestep τ with second-order accuracy. That is, the error Hamiltonian will be of order τ^3 or higher (Saha & Tremaine 1992).

4. IMPLEMENTATION IN AN N-BODY CODE

To test the usefulness of this formulation, we have implemented the above integration algorithm in the parallel gravity code *PKDGRAV* (Stadel 2001) as part of the technique for solving an N-body system in a patch corotating in a Kepler potential (Richardson et al. 2000; Porco et al. 2008; Barnes et al. 2009). This code uses a standard form of leapfrog, where the velocities are first updated by a half step, a *Kick*, then the positions are updated by a full step, a *Drift*, and finally the velocities are given a second half step *Kick*. Only minor changes were needed to implement the above second-order algorithm. The most straightforward way to modify the algorithm is to change the *Kick* routine so that it only includes the terms present in the H_{MIF} Hamiltonian, and modify the *Drift* routine to include the operations of Equations (25) through (28) as well as the standard *Drift* of the positions. However, the existing *Drift* routine in *PKDGRAV* is complicated by the handling of periodic boundary conditions and the search for collisions, so we instead rearranged the operations so that the *Drift* remains a simple linear extrapolation of the positions with constant velocities.

In detail, the modifications are as follows. After the opening *Kick* routine updates the velocities according to H_{MIF}, it calculates the canonical momentum, P_y, and updates $\dot{x}$ (which is equivalent to P_x) using Equation (27). However, $\dot{y}$ is updated to be the sum of the contributions of the free particle Hamiltonian, H_{FP}, the cross term (Equation (26)) applied at the beginning of the *Drift* (for time $\tau/2$), and the cross term applied at the end of the *Drift* (also for a time $\tau/2$):

$$\dot{y} = P_y - \left(\frac{1}{2}\right)2\Omega x - \left(\frac{1}{2}\right)2\Omega(x + \tau\dot{x}). \tag{29}$$

(Also see Equation (33) below.) This $\dot{y}$ along with $\dot{x}$ can now be used to linearly update the positions according to both "Cross" operators and the free particle operator using an essentially unmodified *Drift* routine. Finally, the closing *Kick* again uses Equation (27) to update $\dot{x}$, sets $\dot{y}$ to be $P_y - 2\Omega x$, and updates the velocities according to H_{MIF}. The above algorithm requires storage for a new attribute, P_y, for each particle.

One modification to the *Drift* routine involves the handling of periodic boundary conditions. Often, Hill's equations are integrated with periodic boundaries in the x- and y-directions. If a particle exits the computational volume in the, e.g., positive y-direction, then it is replaced by a particle with the same velocity and x-coordinate on the negative y-boundary. Handling the x-boundaries is a little more complicated because of the shear across the patch: a particle in a circular orbit on the outer boundary is moving slower in the y-direction than a particle on a circular orbit on the inner boundary by an amount $\frac{3}{2}\Omega\Delta x$, where Δx is the width of the patch. This implies an increase in the P_y of the particle of $\frac{1}{2}\Omega\Delta x$. This corresponds to the fact that in a Kepler potential, the circular velocity of an orbit decreases outward, while the angular momentum of a circular orbit increases outward. Hence, when the *Drift* routine detects a particle has crossed the x-boundary, P_y is changed accordingly.

The *Drift* routine also performs collision detection and resolution. Therefore, if a particle's momentum is changed due to a collision, or if a new particle is created either due to merging or fragmentation, the canonical momentum, P_y, needs to be updated to reflect the change. Solving Equation (29) for P_y provides a simple means to calculate a new P_y at any time during the drift. For a merger, the conservation of the total P_y of the bodies involved in the collision could be used to assign a

P_y to the merged particle. No other changes to *PKDGRAV* were needed to implement this algorithm.

In summary, a single timestep of a single particle starting from the state $(x_n, y_n, \dot{x}_n, \dot{y}_n)$ and producing the state $(x_{n+1}, y_{n+1}, \dot{x}_{n+1}, \dot{y}_{n+1})$ is performed as follows:

$$\dot{x}_{n+1/4} = \dot{x}_n - \frac{\tau}{2}\left(\Omega^2 x_n + \frac{\partial\Phi(x_n, y_n)}{\partial x}\right), \quad (30)$$

$$P_y = \dot{y}_n + 2\Omega x_n - \frac{\tau}{2}\frac{\partial\Phi(x_n, y_n)}{\partial y}, \quad (31)$$

$$\dot{x}_{n+1/2} = \dot{x}_{n+1/4} + \tau\Omega P_y, \quad (32)$$

$$\dot{y}_{n+1/2} = P_y - \Omega x_n - \Omega(x_n + \tau\dot{x}_{n+1/2}), \quad (33)$$

$$x_{n+1} = x_n + \tau\dot{x}_{n+1/2}, \quad (34)$$

$$y_{n+1} = y_n + \tau\dot{y}_{n+1/2}, \quad (35)$$

$$\dot{x}_{n+3/4} = \dot{x}_{n+1/2} + \tau\Omega P_y, \quad (36)$$

$$\dot{x}_{n+1} = \dot{x}_{n+3/4} - \frac{\tau}{2}\left(\Omega^2 x_{n+1} + \frac{\partial\Phi(x_{n+1}, y_{n+1})}{\partial x}\right), \quad (37)$$

$$\dot{y}_{n+1} = P_y - 2\Omega x_{n+1} - \frac{\tau}{2}\frac{\partial\Phi(x_{n+1}, y_{n+1})}{\partial y}. \quad (38)$$

Although fractional subscripts are used for intermediate values of the state variables for clarity, the only extra storage needed for this update is for the canonical momentum, P_y. Equations (30)–(33) are implemented in the first *Kick* routine, and Equations (36)–(38) are implemented in the last *Kick* routine, leaving a simple form for the *Drift*, Equations (34) and (35), during which boundary crossing and collision detection is performed.

5. TESTS OF THE METHOD

As a first test of the usefulness of a symplectic scheme for Hill's equations, we used the implementation in *PKDGRAV* to integrate a single particle in a Kepler potential and compare the conserved integrals of the system using our new integrator with those using a standard second-order integration method. The standard method integrates Equations (2) and (3) with the *Kick–Drift–Kick* leapfrog described above, with the modification that velocities are predicted to the end of the timestep using the old accelerations in order to calculate the velocity-dependent part of the force. This is the same algorithm that is used to handle the velocity-dependent forces arising in SPH (Wadsley et al. 2004).

In this case of a single particle $\Phi = 0$, and a physically relevant combination of the integrals is the eccentricity, which in terms of the canonical coordinates can be expressed as

$$e^2 = \frac{1}{R^2\Omega^2}\left(P_x^2 - 4\Omega x P_y + \Omega^2 x^2 + 4P_y^2\right), \quad (39)$$

where R is the radius of the orbit of the patch. Figure 1 shows the relative change in eccentricity in an integration of a particle with an initial $e = 0.001$ for 100 orbits. The dotted line shows this change for the standard integration method when 100 steps
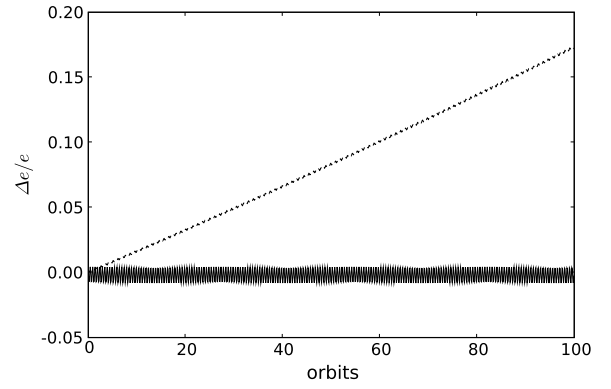

Figure 1. Relative change in eccentricity for a single particle. $\Delta e/e$ is plotted vs. orbit number for a 100 orbit integration of a particle with an initial eccentricity of 0.001. The dotted line shows the evolution using the standard non-symplectic integration algorithm with 100 steps per orbit. The solid line shows the evolution using the symplectic algorithm with 20 steps per orbit. The symplectic algorithm with 100 steps per orbit has a maximum $\Delta e/e$ of 0.00017, so its evolution would be a horizontal line at the resolution of this plot. All integrations were performed with *PKDGRAV* on a single processor.

per orbit are used, while the solid line shows results for our new method with only 20 steps per orbit. The new integrator demonstrates the typical behavior of a symplectic method: the numerical value of the integral of motion oscillates around the true value, and there is no secular drift. An integration with the symplectic integrator using 100 steps per orbit has a maximum $\Delta e/e$ of 0.00017. Comparing this with the maximum $\Delta e/e$ for the integration with 20 steps per orbit, 0.0043, indicates that the error is scaling as τ^2, where τ is the timestep. This is as expected for a second-order integrator. A careful inspection of the figure shows that the dotted line has a slope that is slightly increasing with time, implying that in the standard method, the eccentricity drift in this case grows faster for larger eccentricities.

A somewhat more relevant test of the integrator is following an encounter in the restricted three body approximation. Specifically, we use the *PKDGRAV* implementation to integrate the orbit of a test particle as it comes within a Hill radius of a massive body on a circular orbit. Such a situation is not uncommon in simulations of planetesimal growth: the large bodies are in somewhat circular orbits (BQLR). The accuracy of the integration can be evaluated using the Jacobi integral (Duncan et al. 1989)

$$\Gamma = 3\Omega^2 x^2 - \dot{x}^2 - \dot{y}^2 + \frac{2Gm}{(x^2 + y^2)^{1/2}}, \quad (40)$$

where m is the mass of the massive body. Guided by the end-state configuration of the L_1 simulation in BQLR, we set the mass of the massive body to be 3.78×10^{18} g (300 times the mass of a 1 km planetesimal). The test particle is placed in an orbit such that it comes within one Hill radius of the massive body at aphelion, and the encounter speed at closest approach is given by the rms velocity in the BQLR simulation, 2 m s^{-1}. These parameters imply an eccentricity of 1.6×10^{-4} and a relative difference in semi-major axis $\varepsilon \equiv |a - a_m|/a = 1.69 \times 10^{-4}$, where a_m is the semi-major axis of the massive body. We follow the motion of the test particle starting at perihelion, through the conjunction with the massive body and to the subsequent perihelion. Due to the encounter, the eccentricity of the test body changes by 6.4×10^{-7}. This is somewhat greater than the eccentricity change expected from the mapping formula of Duncan et al. (1989), 1.5×10^{-7}, presumably because this encounter does not satisfy their approximation that $e \ll \varepsilon$.
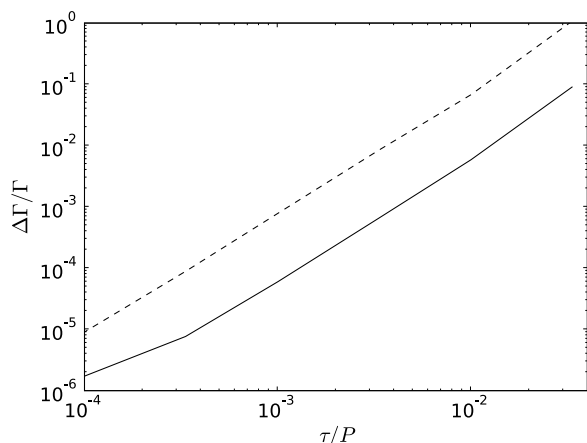
Figure 2. Maximum relative change in the Jacobi constant, Γ, during an encounter with a massive body as a function of step size. A single orbit during which the test body comes within one Hill radius of a massive body is integrated between two successive perihelia. The dotted line shows the maximum change in Γ over this orbit as a function of step size in units of the orbital period for the standard non-symplectic integration. The solid line shows the same quantity for the symplectic algorithm. All integrations were performed with *PKDGRAV* on a single processor.

Figure 2 shows how well Γ is conserved during this encounter as a function of the integration timestep for the symplectic and the non-symplectic integrations. As with the eccentricity in the simple orbit case, the error in the Jacobi constant scales as τ^2. However, for a given timestep the symplectic integration algorithm gives an order of magnitude improvement in the conservation of Γ. At the largest timestep plotted, $\tau = P/30$, the test body moves through the encounter at about $2\,r_H$ per timestep. Hence, reasonably resolving the encounter requires $\tau = P/100$ or smaller.

6. DISCUSSION AND SUMMARY

The choice of algorithm for a numerical simulation is critical to obtaining accurate physical results. This is particularly true for simulations performed over many dynamical times where there is an opportunity for truncation error to buildup in a secular manner. In this case, an algorithm that appears to work well over a few orbits may produce unacceptably incorrect results when used for hundreds of orbits. Using the standard second-order integrator to solve Hill's equations in the context of planetesimal accretion illustrates this problem. Although individual orbits are followed reasonably well with a few hundred steps per orbit, the secular growth in eccentricity shown in Figure 1 could overwhelm any physical changes in the eccentricity distribution of planetesimals in a hundred orbits or so. Faced with this problem, one must either go to a higher order algorithm, which is difficult to implement in a large parallel simulation code, or use much smaller timesteps, which significantly increases the computational expense.

Fortunately, for the case of Hill's equation, we have discovered a second-order symplectic integrator that does not display any secular growth in eccentricity. Moreover, our solution is linear in the time extrapolation of particle positions, permitting efficient collision detection, which is valuable for most large N simulations of planetesimals and planetary rings. The algorithm is derived from the formalism of Saha & Tremaine (1992) where

the Hamiltonian is separated into parts that by themselves are integrable. This separation is in turn made possible via a canonical transformation to coordinates that significantly simplifies the Hamiltonian. The algorithm has been implemented in the scalable parallel *N*-body code *PKDGRAV*, and this code is currently being used for follow-on simulations to those described in BQLR, which simulated the growth of planetesimals over hundreds of orbits.

As described above, our integrator has a fixed timestep which is inefficient for simulations that have to resolve close encounters between bodies where the encounter timescales are a small fraction of the orbital time. For example, in planet formation simulations the encounter timescale is hours compared to an orbital time of one year. Constructing an integrator that can adjust timesteps in order to handle close encounters and yet retains symplectic properties requires care (Duncan et al. 1998; Chambers 1999). In our implementation, we have simply adjusted the timestep of particles that are experiencing a collision. Although this destroys the symplectic properties of the integrator, for the planetesimal simulations in BQLR, a typical particle makes about one hundred orbits before experiencing a collision. Hence, even with the non-symplectic timestep adjustment, our integrator significantly improves the quality of these simulations.

REFERENCES

Barnes, R., Quinn, T. R., Lissauer, J. J., & Richardson, D. C. 2009, Icarus, 203, 626
Chambers, J. E. 1999, MNRAS, 304, 793
Chambers, J. E., & Murison, M. A. 2000, AJ, 119, 425
Duncan, M. J., Levison, H. F., & Lee, M. H. 1998, AJ, 116, 2067
Duncan, M., Quinn, T., & Tremaine, S. 1989, Icarus, 82, 402
Goldreich, P., & Lynden-Bell, D. 1965, MNRAS, 130, 125
Heggie, D. C. 2001, in Proc. Fifty-fourth Scottish Universities Summer School in Physics, The Restless Universe, ed. B. A. Steves & A. J. Maciejewski (Bristol: IOP Publishing & Scottish Universities Summer School in Physics), 109
Hill, G. W. 1878, Am. J. Math., 1, 5
Holman, M. J., & Wisdom, J. 1993, AJ, 105, 1987
Julian, W. H., & Toomre, A. 1966, ApJ, 146, 810
Laskar, J., & Robutel, P. 2001, Celest. Mech. Dyn. Astron., 80, 39
Lee, M. H., & Peale, S. J. 2002, ApJ, 567, 596
Levison, H. F., & Duncan, M. J. 1994, Icarus, 108, 18
Malhotra, R. 1995, AJ, 110, 420
McNeil, D. S., & Nelson, R. P. 2009, MNRAS, 392, 537
Porco, C. C., Weiss, J. W., Richardson, D. C., Dones, L., Quinn, T., & Throop, H. 2008, AJ, 136, 2172
Quinn, T., Katz, N., Stadel, J., & Lake, G. 1997, arXiv:astro-ph/9710043
Richardson, D. C., Quinn, T., Stadel, J., & Lake, G. 2000, Icarus, 143, 45
Rivera, E. J., & Lissauer, J. J. 2000, ApJ, 530, 454
Saha, P., & Tremaine, S. 1992, AJ, 104, 1633
Saha, P., & Tremaine, S. 1994, AJ, 108, 1962
Springel, V. 2005, MNRAS, 364, 1105
Stadel, J. G. 2001, PhD thesis, Univ. Washington
Tanga, P., Weidenschilling, S. J., Michel, P., & Richardson, D. C. 2004, A&A, 427, 1105
Wadsley, J. W., Stadel, J., & Quinn, T. 2004, New Astron., 9, 137
Wisdom, J., & Holman, M. 1991, AJ, 102, 1528
Wisdom, J., & Tremaine, S. 1988, AJ, 95, 925