Particle–Particle Particle–Tree: 
A Direct-Tree Hybrid Scheme for Collisional N-Body Simulations

Shoichi OSHINO,1,2 Yoko FUNATO,4 and Junichiro MAKINO1,2,3
1Department of Astronomical Science, The Graduate University for Advanced Studies (SOKENDAI), 2-21-1 Osawa, Mitaka, Tokyo 181-8588
2Division of Theoretical Astronomy, National Astronomical Observatory of Japan, 2-21-1 Osawa, Mitaka, Tokyo 181-8588
oshino@cfca.jp
3Center for Computational Astrophysics, National Astronomical Observatory of Japan, 2-21-1 Osawa, Mitaka, Tokyo 181-8588
makino@cfca.jp
4Department of General System Studies, Graduate School of Arts and Sciences, The University of Tokyo, 3-8-1 Komaba, Meguro-ku, Tokyo 153-8902
funato@artcompsci.org

(Received 2011 January 19; accepted 2011 May 6)

Abstract

In this paper, we present a new hybrid algorithm for the time integration of collisional N-body systems. In this algorithm, the gravitational force between two particles is divided into short- and long-range terms, using a distance-dependent cutoff function. The long-range interaction is calculated using a tree algorithm, and integrated with a constant-timestep leapfrog integrator. The short-range term is calculated directly and integrated with the high-order Hermite scheme. We can reduce the calculation cost per orbital period from \(O(N^2)\) to \(O(N\log N)\), without significantly increasing the long-term integration error. The results of our test simulations show that close encounters are integrated accurately. Long-term errors of the total energy show random-walk behaviour, because they are dominated by the error caused by the tree approximation.

Key words: methods: n-body simulations — solar system: formation

1. Introduction

For the time integration of collisional N-body systems, such as star clusters and systems of planetesimals, the combination of direct summation for force calculation and the individual timestep algorithm has been the standard method for nearly a half century (Aarseth 1963, 2003). For galactic dynamics and cosmology, fast and approximate methods for force calculation such as the particle-mesh scheme (Hockney & Eastwood 1981), the P^M scheme (Hockney & Eastwood 1981), the tree method (Barnes & Hut 1986), and combinations of PM and tree (Xu 1995; Bagla 2002; Dubinski et al. 2004; Springel 2005; Yoshikawa & Fukushige 2005; Ishiyama et al. 2009) are used. It is not impossible to combine an individual timestep algorithm and a fast and approximate force calculation. For example, McMillan and Aarseth (1993) developed a high-order integrator using individual timesteps combined with the tree algorithm. However, it was difficult to achieve good performance on distributed-memory parallel computers for such a scheme. Fujii et al. (2007) introduced a hybrid of a tree and an individual timestep algorithm, which is designed to handle the evolution of star clusters embedded in the parent galaxy. In their BRIDGE (Bridge is for Realistic Interactions in Dense Galactic Environment) scheme, both the parent galaxy and the star cluster are expressed as N-body systems. The interactions between particles in the star cluster are calculated directly and integrated with the individual timestep scheme, while interactions between particles in the parent galaxy and those between particles in parent galaxy and particles in star clusters are calculated with the tree algorithm and integrated with the leapfrog scheme with shared and constant timesteps.

The BRIDGE scheme is based on the idea of splitting the Hamiltonian of an N-body system into multiple components. The time integration of the tree part is symplectic, and does not generate any secular error. Direct integration of the internal motion of a star cluster is not symplectic, but is treated with high accuracy using high-order integrators. The obvious limitation of the BRIDGE scheme is that it can handle the close encounter of particles in star clusters only. If we want to allow close encounters between particles in the parent galaxy, it goes back to the usual direct summation scheme.

For the time integration of planetary systems, the mixed variable symplectic (hereafter MVS) scheme (Kinoshita et al. 1991; Wisdom & Holman 1991) has become the standard method. For the time integration of almost stable orbits of planets, the MVS scheme is well suited. However, if we want to handle protoplanets or planetesimals, we need to handle their close encounters and collisions. The MVS scheme, however, cannot handle them, since it requires that the timestep is kept constant (Calvo & Sanz-Serna 1993). In order to handle close encounters, several modifications of MVS method have been proposed. One is the hybrid method by Chambers (1999). It handles close encounters with high accuracy by splitting the Hamiltonian into the close interaction and distant interaction, as in the case of the P^M method. It can be used for planetary accretion problems. However, it relies on direct calculations, and its calculation cost scales as \(O(N^2)\). Very recently, Moore, Quillen, and Edgar (2008) applied a general-purpose
computing on graphics processing units (GPGPU) to the hybrid method. In addition, they used the Hermite scheme (Makino & Aarseth 1992) for the integration of the short-range force. Brunini and Viturro (2003) and Brunini et al. (2007) combined the hybrid method and the tree algorithm. Therefore, their method can handle a large number of particles and close encounters. The scheme described in Brunini and Viturro (2003) uses the leapfrog integrator to handle the Hamiltonian for short-range interaction. Therefore, it had to use very small timesteps. The scheme described in Brunini et al. (2007) seems to be improved, but no details of the scheme have been given.

In this paper, we describe a new time-integration algorithm, which combines the strong points of these schemes. This scheme is a combination of the BRIDGE scheme (Fujii et al. 2007) and a hybrid symplectic integrator method (Chambers 1999). We call this scheme Particle–Particle Particle–Tree (hereafter PPPT). In section 2 we overview previous numerical methods. These methods are based on the MVS method. Therefore, we first introduce the MVS method, and then overview the extension of MVS method. In section 3 we describe our new time integration algorithm, PPPT. In section 4, we show the results of test simulations. In this paper, we discuss only simulations of the planet formation process. However, in future work, we will apply PPPT to other collisional systems. A summary and discussions are given in section 5.

2. The Numerical Method

2.1. The Symplectic Integrator

The Hamiltonian of an N-body system is given by

$$H = \sum_{i=1}^{N} \frac{p_i^2}{2m_i} - \sum_{i=1}^{N} \sum_{j=i+1}^{N} \frac{Gm_im_j}{r_{ij}},$$  \hspace{1cm} (1)

where $p_i$ is the momentum of particle $i$, $r_{ij}$ the distance between particles $i$ and $j$, $m_i$ the mass of particle $i$, and $G$ the gravitational constant. The Hamilton’s equation of motion is

$$\frac{df}{dt} = \{f, H\},$$  \hspace{1cm} (2)

where $\{f, H\}$ is the Poisson bracket and $f$ is a canonical variable. We define the differential operator as $Df \equiv \{f, H\}$. The general solution of equation (2) at time $t + \Delta t$ is formally written as

$$f(t + \Delta t) = e^{D\Delta t} f(t).$$  \hspace{1cm} (3)

Equation (3) cannot be solved analytically. We can obtain an approximate solution by dividing the Hamiltonian into multiple parts that can be analytically solved.

In the symplectic integrator, the Hamiltonian is divided into two parts as

$$H = H_A + H_B,$$

$$H_A = -\sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \frac{Gm_im_j}{r_{ij}},$$  \hspace{1cm} (5)

$$H_B = \sum_{i} \frac{p_i^2}{2m_i},$$  \hspace{1cm} (6)

where $H_A$ is the potential energy and $H_B$ is the kinetic energy, and each of them can be solved. The time evolution of $f$ is given by

$$f(t + \Delta t) = e^{D\Delta t} f(t) = e^{D\Delta t(A+B)} f(t),$$  \hspace{1cm} (7)

where $A \equiv \{H_A\}$ and $B \equiv \{H_B\}$ are operators. The exponential in equation (7) can be approximated as

$$e^{D\Delta t(A+B)} = \prod_{i=1}^{k} e^{D\Delta tA} e^{D\Delta B} + O(\Delta t^{n+1}),$$  \hspace{1cm} (8)

where $(a_1, a_2, ..., a_k)$ and $(b_1, b_2, ..., b_k)$ are real numbers, $k$ is the number of stages, and $n$ is the order of approximation. The first term of the right-hand side of equation (8) is a $n$th-order approximation of the left-hand side. The first-order symplectic integrator is given by

$$f(t + \Delta t) = e^{\Delta tA} e^{\Delta tB} f(t) + O(\Delta t^2),$$  \hspace{1cm} (9)

and the second-order symplectic integrator, which is the leapfrog integrator, is given by

$$f(t + \Delta t) = e^{\Delta tA/2} e^{\Delta tB} e^{\Delta tA/2} f(t) + O(\Delta t^3).$$  \hspace{1cm} (10)

The symplectic integrator has the advantage that there is no long-term energy error for the time integration of periodic systems. In the case of a near-Kepler potential, both the semi-major axis and the eccentricity are conserved. One disadvantage of the symplectic method is that high-order schemes are expensive and have rather large local error coefficients. Another disadvantage is that it requires a constant timestep. If we change the timestep following the usual recipes for Runge–Kutta–Fehlberg-type schemes (Fehlberg 1968), the energy is no longer conserved (Skeel & Gear 1992; Calvo & Sanz–Serna 1993). There are a number of proposals of methods for combining the variable timesteps and the good nature of the symplectic schemes. Most of them are based on the idea of splitting the potential into first- and second-varying terms and applying different timesteps. For example, Skeel and Biesiadecki (1994) proposed a method that splits the gravitational force into many components, each of which has finite effective range in the distance. By assigning different timesteps to different components, they effectively realized a variable timestep integration.

The methods discussed below are all based on the idea of finding the way to split the Hamiltonian.

2.2. The Mixed Variable Symplectic Method

Kinoshita, Yoshida, and Nakai (1991) and then Wisdom and Holman (1991) introduced the MVS method for planetary systems. In the symplectic scheme, we split the Hamiltonian into the kinetic and potential energy, so that we obtain analytical solutions for both parts. This division is not unique. As far as each splitted Hamiltonian has an analytic solution, any division can be used. The idea of MVS is to split the Hamiltonian into the Keplerian term $H_{\text{Kep}}$ and interaction term $H_{\text{Int}}$. The time evolution is given by

$$H = H_{\text{Kep}} + H_{\text{Int}},$$  \hspace{1cm} (11)

$$f(t + \Delta t) = e^{\Delta tK/2} e^{\Delta tI} e^{\Delta tK/2} f(t),$$  \hspace{1cm} (12)

where $K$ is the operator defined as $Kf \equiv \{f, H_{\text{Kep}}\}$ and $I$
is the operator defined as $I f \equiv \{ f, H_{\text{Int}} \}$. The MVS method integrates $H_{\text{Int}}$ with the leapfrog integrator and $H_{\text{Kep}}$ by using analytic solution of the Kepler orbit. Thus, the MVS method is expressed as follows:

1. Calculate accelerations due to gravitational interactions between planets at time $t$ and give a velocity kick.
2. Update analytically the positions and velocities from $t$ to $t + \Delta t$ by using the solar gravity.
3. Calculate the accelerations due to gravitational interactions between planets at time $t + \Delta t$ and give a velocity kick.

The advantage of MVS is that only interactions between planets are integrated numerically. The solar gravity is analytically integrated and is accurate up to the round-off error.

2.3. The BRIDGE Code

The BRIDGE code was introduced by Fujii et al. (2007). It was designed for the time integration of star clusters embedded in parent galaxies. This scheme is a combination of the direct and tree schemes, using an idea similar to that of the MVS scheme. The internal interactions of stars in star clusters are integrated by the Hermite integrator with direct summation, while other parts are integrated by the leapfrog integrator and tree scheme.

The BRIDGE scheme divides the Hamiltonian into two parts:

$$H = H_a + H_\beta ,$$
$$H_a = - \sum_{i<j} \frac{G G_i m_{G,j}}{r_{G G,i j}} - \sum_{i=1}^{N_G} \sum_{j=1}^{N_G} \frac{G m_{G,i} m_{C,j}}{r_{G C,i j}},$$
$$H_\beta = \sum_{i=1}^{N_G} \frac{P_{G,i}^2}{2m_{G,i}} + \sum_{i=1}^{N_G} \sum_{j=1}^{N_G} \frac{P_{C,i}^2}{2m_{C,i}} - \sum_{i<j} \frac{G m_{C,i} m_{C,j}}{r_{C C,i j}},$$

where $N_G$ and $N_C$ are the numbers of particles in the parent galaxy and in the star cluster; $m_{G,i}$ and $P_{G,i}$ are the mass and the momentum of particle $i$ in the galaxy; $m_{C,i}$ and $P_{C,i}$ are those of particle $i$ in the star cluster; and $r_{G G},$ $r_{G C},$ and $r_{C C}$ are the distances between two galaxy particles, between one galaxy particle and one star-cluster particle, and between two star-cluster particles, respectively. We can express the time evolution from $t$ to $t + \Delta t$ as

$$f(t + \Delta t) = e^{\frac{1}{2} \Delta t \alpha} e^{\Delta t \beta} e^{\frac{1}{2} \Delta t \alpha} f(t),$$

where $\alpha$ is the operator defined as $\alpha f \equiv \{ f, H_a \}$ and $\beta$ is the operator, defined as $\beta f \equiv \{ f, H_\beta \}$

The BRIDGE code uses the leapfrog scheme for $H_a$, and the fourth-order Hermite scheme for $H_\beta$. Thus, the integration procedure during a tree timestep of $\Delta t$ is done in the following way:

1. Make a tree at time $t$ and calculate the accelerations from all particles on galaxy particles, and from galaxy particles on star-cluster particles.
2. Give a velocity kick for star-cluster particles, and update the velocities of galaxy particles.
3. Integrate positions and velocities of star-cluster particles from $t$ to $t + \Delta t$ by using the Hermite scheme with the individual timestep, and update positions of galaxy particles by making them drift with the constant velocities.
4. Make a new tree at $t + \Delta t$ and calculate the accelerations from all particles to galaxy particles, and from galaxy particles to star-cluster particles.
5. Give a velocity kick for star-cluster particles and update velocities for galaxy particles.

In this scheme, $H_a$ is integrated with the symplectic leapfrog scheme, while $H_\beta$ is integrated with a nonsymplectic Hermite scheme. It combines the fast tree code for the orbital motion of particles in the galaxy and the high-accuracy Hermite scheme for the internal orbital motion of particles in the star cluster without any additional approximation. Thus, this scheme is the first to prove that we can follow the orbital and internal evolution of a star cluster embedded in a galaxy in a fully self-consistent way.

2.4. The MERCURY Code

The MERCURY code (Chambers 1999) splits the Hamiltonian into three parts as

$$H = H_{\text{Kep}} + H_{\text{Int}} + H_{\text{Sun}},$$
$$H_{\text{Kep}} = \sum_{i=1}^{N} \left( \frac{p_i^2}{2m_i} - \frac{G m_i m_{\odot}}{R_i} \right) - \sum_{i<j}^{N} G m_i m_j W(R_{ij}),$$
$$H_{\text{Int}} = - \sum_{i<j}^{N} \frac{G m_i m_j}{R_{ij}} [1 - W(R_{ij})],$$
$$H_{\text{Sun}} = \sum_{i=1}^{N} \frac{p_i^2}{2m_{\odot}},$$

where $P$ and $R$ are the “democratic heliocentric” variables (Duncan et al. 1998), $H_{\text{Int}}$ is the potential energy of the gravitational interactions between particles, except for those undergoing close encounters, $H_{\text{Kep}}$ is the kinetic energy of the particles plus the potential energy of particles undergoing close encounters, $H_{\text{Sun}}$ is the kinetic energy of the Sun, and $W(R_{ij})$ is the changeover function. The modification of MVS is that the potential energy of nearby particles is separated from $H_{\text{Int}}$ and moved to $H_{\text{Kep}}$. The time evolution is described as

$$f(t + \Delta t) = e^{\frac{1}{2} \Delta t \alpha} e^{\Delta t \beta} e^{\frac{1}{2} \Delta t \alpha} f(t),$$

where $I$ is an operator defined as $I f \equiv \{ f, H_{\text{Int}} \}$, $S$ an operator defined as $S f \equiv \{ f, H_{\text{Sun}} \}$ and $K$ an operator defined as $K f \equiv \{ f, H_{\text{Kep}} \}$. This scheme is sometimes called the hybrid scheme. In the actual code (MERCURY), the force of gravitational interactions is split instead of the Hamiltonian for ease of programming. We can write the pairwise force, $F(R_{ij})$, as follows:

$$F(R_{ij}) = F_{\text{close}}(R_{ij}) + F_{\text{distant}}(R_{ij}),$$
$$F_{\text{close}}(R_{ij}) = F(R_{ij}) K(R_{ij}),$$
$$F_{\text{distant}}(R_{ij}) = F(R_{ij}) [1 - K(R_{ij})],$$

where $F_{\text{close}}(R_{ij})$ is the force in the case of close encounters and $F_{\text{distant}}(r_{ij})$ is the remaining force. The relation between $W$ and $K$ is given by
In the hybrid scheme, $H_{\text{Kep}}$ does not have an analytical solution if the potential of a close encounter is not zero. Therefore, $Kf$ is numerically integrated using the Bulirsch–Stoer method (Bulirsch & Stoer 1964; Stoer & Bulirsch 1980). The one-step integration of the hybrid scheme is done in the following way:

1. Apply the velocity kick, $e^{\Delta t/2}$, due to a distant interaction, $H_{\text{Int}}$.
2. Apply the position drift, $e^{\Delta t/2}S$, due to $H_{\text{Sun}}$.
3. Integrate orbits of particles which are under close encounters using Bulirsch–Stoer method, and update positions and velocities of the rest of particles using the Kepler orbit.
4. Give the position drift by $H_{\text{Sun}}$ with a step size of $\Delta t/2$.
5. Give the velocity kick by $H_{\text{Int}}$ with a step size of $\Delta t/2$.

The hybrid method is used for long-term integrations of the outer solar system, restricted three-body problems, and planetary embryos. It gives a good performance in accuracy and speed for small-$N$ systems, but for systems with a large number of particles it becomes expensive.

2.5. The DAEDALUS Code

Brunini et al. (2007) presented a new mixed-variable symplectic tree code for planetesimal dynamics, DAEDALUS. This code is an improved version of the modified tree code described in Brunini and Vituro (2003). Brunini and Vituro (2003) developed the tree code (Barnes & Hut 1986) with two-level timesteps for planetesimal systems. This tree code usually integrates all particles with a constant timestep using the leapfrog integrator. If close encounters occur, it integrates only particles that undergo close encounters with much smaller timesteps. When it integrates close encounters, the timesteps are determined by using equation (2) in Brunini and Vituro (2003), and a new tree is constructed at each step.

The DAEDALUS integrator is a combination of the tree code and a hybrid symplectic integrator method. It splits the Hamiltonian following the description of Chambers (1999). Therefore, the DAEDALUS integrator integrates $H_{\text{Int}}$ with the tree method and integrates $H_{\text{Kep}}$ analytically where there are no close encounters. If close encounters occur, it integrates $H_{\text{Kep}}$ numerically with the Bulirsch–Stoer method. The one-step integration of the DAEDALUS integrator is done in the following way:

1. Make a tree and calculate accelerations from $H_{\text{Int}}$, and then give a velocity kick.
2. Give a position drift by $H_{\text{Sun}}$ with a step size of $\Delta t/2$.
3. Calculate the positions and velocities for particles which are in close encounters, and update the positions and velocities with the Kepler orbit for particles that are not in close encounters.
4. Apply the position drift, $e^{\Delta t/2}S$, due to $H_{\text{Sun}}$.
5. Make a new tree at the next step and calculate the accelerations from $H_{\text{Int}}$, and then give a velocity kick.

The DAEDALUS integrator uses the variable, but shared timestep with particles which are within the effective radius of close encounters. Therefore, when the number of particles in close encounters is large, there is an increase in the cost of calculation. If we use individual timesteps, we can cut the calculation cost significantly. In Table 1, we summarize the characteristics of the previous and present algorithms. Table 1 shows that there was no such an algorithm that has all of the listed desirable properties, except the one we describe in this paper. In this paper, we present a new algorithm which uses the shared timestep for distant interactions, and use the individual timestep for close interactions. Furthermore, the forces due to distant interactions are calculated by using the tree method with a changeover function.

### Table 1. Characteristics of the methods discussed in section 2.

<table>
<thead>
<tr>
<th>Method</th>
<th>Tree method</th>
<th>Distant-based criterion</th>
<th>Variable timestep</th>
<th>Individual timestep</th>
</tr>
</thead>
<tbody>
<tr>
<td>MVS</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>MERCURY</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>BRIDGE</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>DAEDALUS</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>This paper</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

*“Individual timestep” means that each particle has its own time and timestep, and thus even particles interacting with the short-range force can have different timesteps.*

$$W(R) = R \int_{R}^{\infty} \frac{K(x)}{x^2} dx$$

(25)
where $H_{\text{Hard}}$ contains the kinetic energy of all particles and the potential energy of near-neighbours, and $H_{\text{Soft}}$ contains the potential energy of all other pairs of particles. We treat the solar gravity as the force caused by a fixed potential in this paper. We therefore do not have $H_{\text{Sun}}$ here and use the heliocentric coordinates. We can express the time evolution from $t$ to $t + \Delta t$ as

$$f(t + \Delta t) = e_{\Delta t} H^i e_{\Delta t} H C^j f(t).$$

(32)

The changeover function, $K$, splits the force of the gravitational attraction between particles into contributions of close encounters and others. Thus, $H_{\text{Hard}}$ changes rapidly and $H_{\text{Soft}}$ changes slowly. In this paper, we use two types of changeover functions. One is the fourth-order spline function, which was introduced by Abe et al. (1986), and is given as

$$K(r_{ij}) = \left( \frac{\sin X}{X} \right)^5,$$

(33)

where $X = \pi r_{ij}/r_{\text{cut}}$ and $r_{\text{cut}}$ is a scaling radius. Note that $K(r_{ij})$ becomes zero where $r_{ij} \geq r_{\text{cut}}$ (see figure 1). The other is the function that was first introduced by Levison and Duncan (2000). It is given by

$$K(r_{ij}) = \begin{cases} 1 & \text{if } Y \geq 1, \\ 10Y^6 - 15Y^8 + 6Y^{10} & \text{if } 0 < Y < 1, \\ 0 & \text{if } Y \leq 0, \end{cases}$$

(34)

where $Y = (r_2 - r_{ij})/(r_2 - r_1)$. Hereafter we call it the DLL function (Duncan et al. 1998). In figure 1, $r_1/r_{\text{cut}} = 0.4$ and $r_2/r_{\text{cut}} = 0.6$. We call $r_2$ the cutoff radius of the DLL function.

In this paper we regard the gravitational field of the Sun as an external potential. This treatment is satisfactory for the study of planet formation process of Earth-type planets, because planetesimals do not perturb the Sun strongly. The total mass of planetesimals is much smaller than the mass of the Sun, and planetesimals are distributed almost uniformly around the Sun.

We integrated $H_{\text{Soft}}$ using the leapfrog scheme and the tree method with a constant timestep, $\Delta t$. We integrated $H_{\text{Hard}}$ using the fourth-order Hermite method with the block timestep. For the timestep criterion, we used a slightly modified version of the “standard” criterion (Makino & Aarseth 1992; Aarseth 2003). The standard criterion is given by

$$\Delta t_0 = \eta \frac{|a_i| (|a_i|^{(2)} + |\dot{a}_i|)^2}{|\dot{a}_i| (|a_i|^{(3)} + |\ddot{a}_i|)^2},$$

(35)

where $\eta$ is the accuracy parameter. The new timestep criterion we used is given as

$$\Delta t_0 = \eta \frac{|a_i|^2 G m_i}{r_{\text{cut}}^3},$$

(36)

$$a_0 = \alpha \frac{G m_i}{r_{\text{cut}}^3}.$$

(37)

Here, $a_0$ is a constant introduced to prevent the timestep from becoming unnecessarily small when $|a_i|$ is small, and $\alpha$ is a parameter that controls the size of the timestep. Here, $a_i$, $a_i^{(2)}$, and $a_i^{(3)}$ are its first, second, and third time derivatives, respectively, and $m_i$ is the mass of the particle. With equation (35), the timestep becomes unnecessarily small if there is just one particle inside radius $r_{\text{cut}}$ of one particle and $r_{ij} \leq r_{\text{cut}}$. To illustrate this problem, consider the case where one particle moves away radially with a constant velocity $v$. High-order derivatives of the force of this particle is given as follows:

$$a = FK,$$

(38)

$$\dot{a} = (F'K + FK')v,$$

(39)

$$a^{(2)} = (F''K + 2F'K' + FK'')v^2 + F'v^2,$$

(40)

$$a^{(3)} = (F'''K + 3F''K' + 3F'K'' + FK''' + FK''')v^3 + F'v^2 \frac{d}{dx},$$

(41)

for the equation (35), where $F$ is the gravitational force of a particle and $K$ is the changeover function. The solid and dashed curves show the fourth-order spline function and the DLL function, respectively. These functions use $r_1/r_{\text{cut}} = 0.4$ and $r_2/r_{\text{cut}} = 0.6$.
\[ \Delta t = \eta \sqrt{\frac{45Z^4 F^2 + O(Z^5)}{700Z^3 F^2 + O(Z^5)}}. \]

\[ = \eta \frac{9 + O(Z)}{140 + O(Z)}. \]

\[ = \eta \frac{9}{140} \left( \frac{1}{2} \right) + O(Z). \]

Equation (52) shows that the timestep approaches to zero as \( r_{ij} \) approaches to \( r_{\text{cut}} \). This behaviour is clearly undesirable, since there is no need to reduce the timestep for the neighbour force, when the neighbour force itself is small. The reason why criterion (35) gives zero step size is, as we can see from equations (47)–(49), that \( |a| \) approaches to zero faster than its high order derivatives. However, this is due to the cutoff by the changeover function, and the actual physical acceleration by the particle just inside radius \( r_{\text{cut}} \) is on the order, \( m_j / r_{\text{cut}}^2 \). In order to avoid this unnecessarily small timestep, we introduce \( a_0 \) in equation (37). By doing so, we make criterion (37) to give a timestep that is accurate relative to the absolute strength of the force, itself, before the changeover function is applied. This timestep criterion does not use the gravitational force of the Sun. In other words, the timestep is determined purely by the forces of nearby particles. If we included the solar gravity, the original criterion could lead to unnecessarily small timesteps, since the solar gravity is much larger than the forces of neighbour particles.

Our scheme is summarized as follows:

1. Make a tree at time \( t \) and calculate accelerations due to \( H_{\text{Soft}} \).
2. Give a velocity kick.
3. Integrate positions and velocities from \( t \) to \( t + \Delta t \) using the Hermite scheme with the block timestep and \( H_{\text{Hard}} \).
4. Go back to step 1.

When we integrate \( H_{\text{Hard}} \), we use a list of neighbours for particles to save the calculation time. If we do not use this neighbour list, we have to calculate forces of all particles, and the calculation cost becomes \( O(N^2) \). We construct the neighbour list of particle \( i \) by selecting particles within distance \( r_{nl} \) from particle \( i \) at time \( t \). Here, \( r_{nl} \) must be sufficiently larger than \( r_{\text{cut}} \), so that particles outside radius \( r_{nl} \) do not enter the sphere of radius \( r_{\text{cut}} \) during one timestep. In this paper, we use \( r_{nl} \geq r_{\text{cut}} + 3\Delta t \sigma \), where \( \sigma \) is the velocity dispersion. In order to find neighbours fast, we use a uniform 2-D grid with the grid size smaller than \( r_{\text{cut}} \). To summarize our neighbour finding way, we first assign all particles to cells. We then look over the neighbouring cells that are within a distance of \( r_{nl} \) from a particle. The particles in the neighbouring cells are its neighbours.

4. Accuracy and Performance

In this section, we present the result of test calculations for the accuracy and performance of our new algorithm. We adopted the distribution of planetesimals following the Hayashi model for test calculations. The surface density at 1 AU is 10 g cm\(^{-2}\). The unit mass, the gravitational constant \( G \), and the unit length are normalized to one solar mass, one, and 1 AU, respectively. For most of the tests, we used 10000 equal-mass particles distributed randomly between the radii of 0.9 and 1.0 AU. Their mass was \( 1.45 \times 10^3 \) g, and their velocities followed the Rayleigh distribution with \( \langle \epsilon \rangle = 5r_H \) and \( \langle i \rangle = 2.5r_H \), where \( \langle \epsilon \rangle \) and \( \langle i \rangle \) are the dispersions of the eccentricity and inclination. The initial radius of particles was 226 km, which corresponds to a density of 3 g cm\(^{-3}\). The physical collisions were handled under the assumption of perfect accretion. The radius of the collision product was determined so as to keep the density unchanged. Unless otherwise specified, all test calculations were for 10 orbital periods.

Figures 2 and 3 show the relative energy error of the system, \(|E - E_0| / |E_0|\), where \( E_0 \) is the energy of the system at time 0, and the number of direct gravitational interactions per one tree timestep per one particle, \( N_{\text{direct}} \), as a function of the cutoff radius, \( r_{\text{cut}} \), for the case of the spline changeover function [equation (33)]. The cutoff radius, \( r_{\text{cut}} \), is normalized by the Hill radius, \( r_H \), at 1 AU. In figure 2, we plot the largest energy error during the time integration for 10 yr (10 periods). We use \( \eta = 0.05 \), timesteps \( \Delta t = 0.040, 0.020, 0.010, \) and 0.005 yr, and opening angles \( \theta = 0.5 \) and 0.1. The relative energy error is practically independent of \( r_{\text{cut}} \) if \( r_{\text{cut}} / r_H > 3 \) and \( \Delta t < 0.020 \) yr. The number of mutual interactions is proportional to the square of the cutoff radius (figure 3). Since the scale height of the disk is \( \sim 5r_H \), it is in most cases smaller than \( r_{\text{cut}} \). Therefore the number of particles is proportional to the square of the radius. Figures 2 and 3 show that by using \( r_{\text{cut}} / r_H \sim 3 \), we can attain high accuracy with a very small value of \( N_{\text{direct}} \). On the other hand, the increase in the calculation cost is fairly small, even when we use such a very large \( r_{\text{cut}} / r_H \) as 50, because the force calculation using the tree is more expensive. In the case of \( \theta = 0.5 \), the energy error is lower-bounded at \( 10^{-8} \), while with \( \theta = 0.1 \), the error can be reduced to \( 10^{-10} \). In this case, using \( \Delta t = 0.040 \) yr resulted in a rather large error as shown in figure 2. This behaviour can be understood by looking at the time evolution of the error. Figure 4 shows the time evolution of the error for \( \Delta t = 0.040 \) yr and 0.010 yr. The behaviour of the error shows a quasi-periodic behaviour with a period \( \sim 1 \) yr for \( \Delta t = 0.040 \) yr. In figure 2, the relative energy error for \( \Delta t = 0.040 \) yr and \( r_{\text{cut}} / r_H = 20 \) is larger than that for \( \Delta t = 0.040 \) yr and \( r_{\text{cut}} / r_H = 7 \). This is because the random energy error from tree scheme with a large timestep is dominant for \( \Delta t = 0.040 \) yr. Thus, this peculiar behaviour occurred.

Figures 5 and 6 show the energy error for the case of the DLL cutoff function, with the inner radius being \( r_I / r_H = 1 \) and 10. We can see that the behaviour of the error is quite similar to that in the case of the spline cutoff, and independent of the choice of \( r_I \).

Figures 7 and 8 show the relative energy error and the number of interactions per one particle in the tree part as a function of the opening angle \( \theta \). In figure 7, the energy error shows the power-law dependence as \( \propto \theta^{-5} \) for the case of \( \Delta t = 0.005 \) yr. On the other hand, for \( \Delta t = 0.040 \) yr, the error does not go below \( 10^{-9} \) for small values of \( \theta \). This is because, for \( \Delta t = 0.040 \) yr, the truncation error of the integrator becomes larger than the error due to force approximation for \( \theta = 0.1 \). In figure 8, the number of interactions per
one particle is about one order of magnitudes smaller than that for direct calculation, for \( \theta = 0.1 \). In figure 8, we can see that the dependence of \( N_{\text{tree}} \) to \( \theta \) is rather weak. In the case of stellar systems, the calculation cost is proportional to \( \theta^{-2-\gamma} \) (Makino 1991). In our experiment the dependence is \( \sim \theta^{-1} \). This is because the distribution of particles is a thin and narrow ring. Figure 9 shows the relative energy error as a function of the opening angle for the case of the DLL function again; the behaviour of the error is similar to the case of the spline function.

Figure 10 shows the relative energy error as a function of the size of the timestep for \( H_{\text{soft}} \). In the case of \( \theta = 0.1 \) and \( \Delta t < 0.010 \text{yr} \), the error is dominated by that from the tree approximation, and becomes independent of \( \Delta t \). If we want to keep the error in 10yr (10 periods) to be less than \( 10^{-9} \), a pair of \( \theta \sim 0.2 \) and \( \Delta t \sim 0.02 \) is probably a good choice. In realistic
calculations $10^{-9}$ in 10 orbits is probably satisfactory, though how small the error should be is a difficult question. At least, the error of order $10^{-9}$ is smaller than that of the energy change of planetesimals due to gas drag and collisional damping. We do not show the result for the DLL cutoff here. We calculated it by using same parameters as those for the DLL cutoff, and confirmed that the result is essentially the same as that for the spline cutoff in figure 10.

Figures 11 and 12 show the relative energy error and the calculation cost of neighbour force, $N_{\text{direct}}$, as a function of the accuracy parameter $\eta$. Other parameters are $\theta = 0.1$, $r_{\text{cut}}/r_H = 10$, and $\Delta t = 0.0050 \text{yr}$. Figure 11 shows that the energy error is practically independent of a choice of $\alpha$. On the other hand, in figure 12, a small $\alpha$ results in the increase of $N_{\text{direct}}$. In practice, a pair of $\eta = 0.1$ and $\alpha = 1$ seems to be a good choice.
Fig. 7. Relative energy error of the system with the spline function plotted against the opening angle $\theta$. The open squares, open circles, filled squares, and filled circles show the results with $r_{\text{cut}}/r_H = 10$ and $\Delta t = 0.04$ yr, $r_{\text{cut}}/r_H = 50$ and $\Delta t = 0.04$ yr, $r_{\text{cut}}/r_H = 10$ and $\Delta t = 0.005$ yr, and $r_{\text{cut}}/r_H = 50$ and $\Delta t = 0.005$ yr, respectively. The filled squares and circles are indistinguishable.

Fig. 8. Number of tree gravitational interactions per one tree timestep per one particle with the spline function plotted against the opening angle $\theta$. The open squares, open circles, filled squares, and filled circles show the results with $r_{\text{cut}}/r_H = 10$ and $\Delta t = 0.04$ yr, $r_{\text{cut}}/r_H = 50$ and $\Delta t = 0.04$ yr, $r_{\text{cut}}/r_H = 10$ and $\Delta t = 0.005$ yr, and $r_{\text{cut}}/r_H = 50$ and $\Delta t = 0.005$ yr, respectively. The four results are practically indistinguishable.

Fig. 9. Relative energy error of the system with the DLL function plotted against the opening angle $\theta$. The left panel shows the results with $r_1/r_H = 1$ and the open squares, open circles, filled squares, and filled circles show the results with $r_2/r_H = 3$ and $\Delta t = 0.04$ yr, $r_2/r_H = 10$ and $\Delta t = 0.04$ yr, $r_2/r_H = 3$ and $\Delta t = 0.005$ yr, and $r_2/r_H = 10$ and $\Delta t = 0.005$ yr, respectively. The right panel shows the results with $r_1/r_H = 10$ and the open squares, open circles, filled squares, and filled circles show the results with $r_2/r_H = 15$ and $\Delta t = 0.04$ yr, $r_2/r_H = 50$ and $\Delta t = 0.04$ yr, $r_2/r_H = 15$ and $\Delta t = 0.005$ yr, and $r_2/r_H = 50$ and $\Delta t = 0.005$ yr, respectively. The filled squares and circles are indistinguishable.
Figure 13 shows the long-term variation of the relative energy error. The calculation is done with the opening angle $\theta = 0.5$, the cutoff radius $r_{\text{cut}}/r_H = 10$, and the timestep $\Delta t = 0.0050 \text{ yr}$. In this case, the energy error reaches $\sim 7.5 \times 10^{-8}$ after the time integration for $10^4 \text{ yr}$ ($10^4$ orbital periods), while it is $9.8 \times 10^{-8}$ for 10 yr (10 periods). In other words, the energy error grows to be 10 times larger as the integration time becomes 1000 times longer. This shows that the growth of the energy error is stochastic, like a random walk. It means that the error is mainly caused by the force error of the tree scheme (Barnes & Hut 1989). The growth of the energy error is, therefore, expected to be slow, and the error is small enough even after long calculations.

Figure 14 shows the calculation time per one tree timestep as a function of the total number of particles in system $N$. The calculation is done with a opening angle of $\theta = 1$, a cutoff radius of $r_{\text{cut}}/r_H = 5$ and a timestep of $\Delta t = 0.0050 \text{ yr}$. We used the Intel(R) Core(TM)2 Quad CPU Q6600 (2.4 GHz).
Fig. 14. Calculation time per 0.0050 yr plotted against a function of number of particles. The crosses and squares show the results of PPPT and fourth-order Hermite scheme, respectively.

It shows that the calculation time increases as \( O(N \log N) \). Therefore, we reduce the calculation cost from \( O(N^2) \) to \( O(N \log N) \).

Figure 15 shows the number of tree interactions, \( N_{\text{tree}} \) per particle as a function of \( N \). We can see that \( N_{\text{tree}} \) is roughly proportional to \( O(\log N) \).

5. Summary and Discussion

We have developed a new hybrid \( N \)-body simulation algorithm for simulating collisional \( N \)-body systems. This new scheme is constructed by combining the tree and direct schemes using a hybrid integrator. The results of test simulations involving the evolution of a planetesimal system show that our new scheme PPPT can drastically reduce the calculation cost, to a level comparable to the cost of a tree scheme with a constant timestep, while keeping the accuracy sufficient for realistic simulations.

In principle, our scheme can be used for collisional systems other than planetary systems, such as globular clusters and stars around a supermassive black hole in the galactic center. We will show the results of simulations of such systems using our scheme in the future.

We are grateful to Tomoaki Ishiyama, Masaki Iwasawa, Michiko Fujii, Kuniaki Koike, Keigo Nitadori, and Yusuke Tsukamoto for fruitful discussions. This work was partially supported by the Research Fund for Students (2009) of the Department of Astronomical Science, the Graduate University for Advanced Studies, and by Japan Society for the Promotion of Science, KAKENHI (grant numbers 21244020 and 21220001).

References

Barnes, J., & Hut, P. 1986, Nature, 324, 446
Bulirsch, R., & Stoer, J. 1964, Numer. Math., 6, 413
Ishiyama, T., Fukushige, T., & Makino, J. 2009, PASJ, 61, 1319
Skeel, R. D., & Gear, C. W. 1992, Physica D, 60, 311
Yoshikawa, K., & Fukushige, T. 2005, PASJ, 57, 849