THE ASTROPHYSICAL JOURNAL, 540:969–982, 2000 September 10

MONTE CARLO SIMULATIONS OF GLOBULAR CLUSTER EVOLUTION. I. METHOD AND TEST CALCULATIONS

KRITEN J. JOSHI,[1] FREDERIC A. RASIO,[2,3] AND SIMON PORTEGIES ZWART[4,5]
Department of Physics, Massachusetts Institute of Technology
Received 1999 September 6; accepted 2000 March 29

ABSTRACT

We present a new parallel supercomputer implementation of the Monte Carlo method for simulating the dynamical evolution of globular star clusters. Our method is based on a modified version of Hénon's Monte Carlo algorithm for solving the Fokker-Planck equation. Our code allows us to follow the evolution of a cluster containing up to 5×10^5 stars to core collapse in $\lesssim 40$ hours of computing time. In this paper we present the results of test calculations for clusters with equal-mass stars, starting from both Plummer and King model initial conditions. We consider isolated as well as tidally truncated clusters. Our results are compared to those obtained from approximate, self-similar analytic solutions, from direct numerical integrations of the Fokker-Planck equation, and from direct N-body integrations performed on a GRAPE-4 special-purpose computer with $N = 16384$. In all cases we find excellent agreement with other methods, establishing our new code as a robust tool for the numerical study of globular cluster dynamics using a realistic number of stars.

Subject headings: celestial mechanics, stellar dynamics — globular clusters: general — methods: *n*-body simulations — methods: numerical

1. INTRODUCTION

The dynamical evolution of dense star clusters is a problem of fundamental importance in theoretical astrophysics, but many aspects of the problem have remained unresolved in spite of years of numerical work and improved observational data. On the theoretical side, some key unresolved issues include the role played by primordial binaries and their dynamical interactions in the overall cluster dynamics and in the production of exotic sources (Hut et al. 1992), and the importance of tidal shocking for the long-term evolution and survival of globular clusters in the Galaxy (Gnedin, Lee, & Ostriker 1999). On the observational side, we now have many large data sets providing a wealth of information on blue stragglers, X-ray sources and millisecond pulsars, all found in large numbers in dense clusters (e.g., Bailyn 1995; Camilo et al. 2000; Piotto et al. 1999). Although it is clear that these objects are produced at high rates through dynamical interactions in the dense cluster cores, the details of the formation mechanisms, and in particular the interplay between binary stellar evolution and dynamical interactions, are far from understood.

1.1. *Overview of Numerical Methods*

Following the pioneering work of Hénon (1971a, 1971b), many numerical simulations of globular cluster evolution were undertaken in the early 1970s, by two groups, at Princeton and Cornell, using different Monte Carlo methods, now known as the "Princeton method" and the "Cornell method" (see Spitzer 1987 for an overview of the methods). In the Princeton method, the orbit of each star is integrated numerically, while the diffusion coefficients for the change in velocity $\Delta \boldsymbol{v}$ and $(\Delta v)^2$ (which are calculated analytically) are selected to represent the average perturbation over an entire orbit. Energy conservation is enforced by requiring that the total energy be conserved in each radial region of the cluster. The Princeton method assumes an isotropic, Maxwellian velocity distribution of stars to compute the diffusion coefficients, and hence does not take in to account the anisotropy in the orbits of the field stars. One advantage of this method is that, since it follows the evolution of the cluster on a dynamical timescale, it is possible to follow the initial "violent relaxation" phase more easily. Unfortunately, for the same reason, it also requires considerably more computing time compared to other versions of the Monte Carlo method. In the Cornell method, also known as the "Orbit-averaged Monte Carlo method", the changes in energy E and angular momentum J per unit time (averaged over an orbit) are computed analytically for each star. Hence, the time consuming dynamical integration of the orbits is not required. In addition, since the diffusion coefficients are computed for both ΔE *and* ΔJ, the Cornell method does take in to account the anisotropy in the orbits of the stars. The "Hénon method" is a variation of the Cornell method, in which the velocity perturbations are computed by considering an encounter between pairs of neighboring stars. This also allows the local two-dimensional phase space distribution $f(E, J)$ to be sampled correctly. Our code is based on a modified version of Hénon's method. We have modified Hénon's algorithm for determining the time step and computing the representative encounter between neighboring stars. Our method allows the time step to be made much smaller in order to resolve the dynamics in the core more accurately. We describe the basic method and our modifications in more detail below in § 2.

[1] 6-218M MIT, 77 Massachusetts Avenue, Cambridge, MA 02139; kjoshi@alum.mit.edu.

[2] 6-201 MIT, 77 Massachusetts Avenue, Cambridge, MA 02139; rasio@mit.edu.

[3] Alfred P. Sloan Research Fellow.

[4] Boston University, 725 Commonwealth Avenue, Boston, MA 02215; spz@komodo.bu.edu.

[5] Hubble Fellow.

The Monte Carlo methods were first used to study the development of the gravothermal instability (Spitzer & Hart 1971a, 1971b; Hénon 1971b) and to explore the effects of a massive black hole at the center of a globular cluster (Lightman & Shapiro 1977). In those early studies, the available computational resources limited the number of particles used in the Monte Carlo simulations to $\lesssim 10^3$. Since this is much smaller than the real number of stars in a globular cluster ($N \sim 10^5$–10^6), each particle in the simulation represents effectively a whole spherical shell containing many stars, and the method provides no information about individual objects and their dynamical interactions. More recent implementations have used up to $\sim 10^4$–10^5 particles and have established the method as a promising alternative to direct N-body integrations (Stodólkiewicz 1986; Giersz 1998). Monte Carlo simulations have also been used to study specific interaction processes in globular clusters, such as tidal capture (Di Stefano & Rappaport 1994), interactions involving primordial binaries (Hut, McMillan, & Romani 1992a) and stellar evolution (Portegies Zwart et al. 1997). However, in all these studies the background cluster was assumed to have a *fixed structure* , which is clearly not realistic. The main goal of our study is to perform Monte Carlo simulations of cluster dynamics treating both the cluster itself and all relevant interactions self-consistently, including all dynamical interactions involving primordial binaries. This idea is particularly timely because the latest generation of parallel supercomputers now makes it possible to do such simulations for a number of objects equal to the actual number of stars in a globular cluster. Using the correct number of stars in a cluster simulation ensures that the relative rates of different dynamical processes (which all scale differently with the number of stars) are correct. This is crucial if many different dynamical processes are to be incorporated, as we plan to do in this study.

In addition to Monte Carlo and N-body simulations, a new method was developed, mainly by Cohn and collaborators, based on the direct numerical integration of the orbit-averaged Fokker-Planck equation (Cohn 1979, 1980; Statler, Ostriker, & Cohn 1987; Murphy & Cohn 1988). Unlike the Monte Carlo methods, the direct Fokker-Planck method constructs the (smooth) distribution function of the system on a grid in phase space, effectively providing the $N \rightarrow \infty$ limit of the dynamical behavior. The original formulation of the method used a two-dimensional phase space distribution function $f(E, J)$ (Cohn 1979). However, the method was later reduced to a one-dimensional form using an isotropized distribution function $f(E)$ (Cohn 1980). The reduction of the method to one dimension speeded up the calculations significantly. In addition, the use of the Chang & Cooper (1970) differencing scheme provided much better energy conservation compared to the original two-dimensional method. The one-dimensional method provided very good results for isolated clusters, in which the effects of velocity anisotropy are small. The theoretically predicted emergence of a power-law density profile in the late stages of evolution for isolated single-component systems has been clearly verified using this method (Cohn 1980). Calculations that include the effects of binary interactions, including primordial binaries, have also allowed the evolution to be followed beyond core collapse (Gao et al. 1991). However, results obtained using the one-dimensional method showed substantial disagreement with N-body results for tidally truncated clusters, in which the evaporation rate is dramatically affected by the velocity anisotropy. Ignoring the velocity anisotropy led to a significant overestimate of the evaporation rate from the cluster, resulting in shorter core-collapse times for tidally truncated clusters (Portegies Zwart et al. 1998). A recent implementation of the Fokker-Planck method by Drukier et al. (1999) has extended the algorithm to allow a two-dimensional distribution function, while also improving the energy conservation. A similar two-dimensional method has also been developed by Takahashi (1995, 1996, 1997). The new implementations produce much better agreement with N-body results (Takahashi & Portegies Zwart 1998), and can also model the effects of mass loss due to stellar evolution (Takahashi & Portegies Zwart 2000), as well as binary interactions (Drukier et al. 1999).

For many years direct N-body simulations were limited to systems with $N \lesssim 10^3$ stars. New, special-purpose computing hardware such as the GRAPE (Makino et al. 1997) now make it possible to perform direct N-body simulations with up to $N \sim 10^5$ single stars (Hut & Makino 1999), but the inclusion of a significant fraction of primordial binaries in these simulations remains prohibitively expensive. The large dynamic range of the orbital timescales of the stars in the cluster presents a serious difficulty for N-body simulations. The orbital timescales can be as small as the periods of the tightest binaries. The direct integration of stellar orbits is especially plagued by this effect. These difficulties are overcome using techniques such as individual integration time steps, and various schemes for regularizing binaries (see, e.g., Aarseth 1998 for a review). These shortcuts introduce specific selection effects, and complicate code development considerably. Instead, in the Monte Carlo methods, individual stellar orbits are represented by their constants of the motion (energy E and angular momentum J for a spherical system) and perturbations to these orbits are computed periodically on a time step that is a fraction of the relaxation time. Thus the numerical integration proceeds on the natural timescale for the overall dynamical evolution of the cluster. Note also that, because of exponentially growing errors in the direct integration of orbits, N-body simulations, just like Monte Carlo simulations, can only provide a statistically correct representation of cluster dynamics (Goodman, Heggie, & Hut 1993; Hernquist, Hut, & Makino 1993).

A great advantage of the Monte Carlo method is that it makes it particularly easy to add more complexity and realism to the simulations one layer at a time. The most important processes that we will focus on initially will be stellar evolution and mass loss through a tidal boundary. Interactions of single stars with primordial binaries, binary-binary interactions, stellar evolution in binaries, and a detailed treatment of the influence of the Galaxy, including tidal shocking of the cluster when it passes through the Galactic disk, will be incorporated subsequently.

Recent improvements in algorithms and available computational resources have allowed meaningful comparisons between the results obtained using different numerical methods (see for example the “collaborative experiment” by Heggie et al. 1998). However, there still remain substantial unresolved differences between the results obtained using various methods. For example, the lifetimes of clusters computed recently using different methods have been found to vary significantly. Lifetimes of some clusters computed using direct Fokker-Planck simulations by Chernoff

& Weinberg (1990) are up to an order of magnitude shorter than those computed using N-body simulations and a more recent version of the Fokker-Planck method (Takahashi & Portegies Zwart 1998). It has been found that, in many cases, the differences between the two methods can be attributed to the lack of an appropriate discrete representation of the cluster in the Fokker-Planck simulations. This can lead to an overestimate of the mass-loss rate from the cluster, causing it to disrupt sooner. Recently, new calibrations of the mass loss in the Fokker-Planck method (Takahashi & Portegies Zwart 2000) that account for the slower mass loss in discrete systems, has led to better agreement between the methods. The limitation of N-body simulations to small N (especially for clusters containing a large fraction of primordial binaries) makes it particularly difficult to compare the results with Fokker-Planck calculations, which are effectively done for very large N (Portegies Zwart et al. 1998; Heggie et al. 1998). This gap can be filled very naturally with Monte Carlo simulations, which can be used to cover the entire range of N not accessible by other methods.

1.2. *Astrophysical Motivation*

The realization over the last 10 years that primordial binaries are present in globular clusters in dynamically significant numbers has completely changed our theoretical perspective on these systems (see e.g., the review by Hut et al. 1992b). Most importantly, dynamical interactions between hard primordial binaries and other single stars or binaries are now thought to be the primary mechanism for supporting a globular cluster against core collapse (McMillan, Hut, & Makino 1990, 1991; Gao et al. 1991). In addition, exchange interactions between primordial binaries and compact objects can explain very naturally the formation of large numbers of X-ray binaries and recycled pulsars in globular cluster cores (Sigurdsson & Phinney 1995; Davies & Hansen 1998; Portegies Zwart et al. 1997). Previously, it was thought that primordial binaries were essentially nonexistent in globular clusters, and so other mechanisms such as tidal capture and three-body encounters had to be invoked in order to form binaries dynamically during core collapse. However, these other mechanisms have some serious problems, and are much more likely to result in mergers than in the formation of long-lived binaries (Chernoff 1996; Kochanek 1992; Kumar & Goodman 1996).

Hubble Space Telescope (*HST*) observations have provided direct constraints on primordial binary fractions in clusters. The binary fraction is a key input parameter for any realistic study of cluster dynamics. For example, the recent observation of a broadened main sequence in NGC 6752, based on *HST* PC images of its core, suggest that the binary fraction is probably in the range 15%–38% in the inner core (Rubenstein & Bailyn 1997).

Despite the fact that binaries play a crucial role in the late phases of evolution of a cluster, the overall evolution of a binary population within a cluster, and its direct implications for the formation rate of observable binaries and blue stragglers remains poorly understood. In addition, the relative importance of binaries in a cluster, like many other physical processes, may depend on the actual size (N) of the cluster. This makes it difficult to extend results obtained from smaller N-body simulations to realistic globular cluster models. When the initial primordial binary fraction is below a certain critical value, a globular cluster core can run out of binaries before the end of its lifetime, i.e., before being evaporated in the tidal field of the Galaxy (McMillan & Hut 1994). Without the support of binaries, the cluster will undergo a much deeper core collapse and so-called gravothermal oscillations (Sugimoto & Bettwieser 1983; Breeden, Cohn, & Hut 1994; Makino 1996). At maximum contraction, the core density may increase by many orders of magnitude, leading to greatly enhanced interaction rates. Our new Monte Carlo code will allow us to follow the evolution of a cluster through this phase, including in detail the dynamical interactions between the $\sim 10^3$ objects in the core.

Of particular interest is the possibility that successive collisions and mergers of MS stars might lead to a *runaway process*. The recent *HST* observations of stellar cusps in the cores of M15 (Guhathakurta et al. 1996; Sosin & King 1997) and NGC 6624 (Sosin & King 1995) have generated renewed interest in the possibility of massive black holes in globular clusters. The most significant unresolved theoretical issue concerns the manner in which such a black hole could form in a dense cluster. One of the likely routes, which we plan to examine with our simulations, is via the collisions and mergers of main-sequence stars, leading to the runaway build-up of a massive object and its eventual gravitational collapse (Portegies Zwart et al. 1999).

A very significant effect of the galactic environment on a cluster is the gravitational shock heating of the cluster due to passages close to the bulge and through the disk. When a cluster passes through the Galactic disk, it experiences a time-varying gravitational force that pulls the cluster toward the equatorial plane. The net effect of the shock is to induce an increase in the average energy of the stars, causing the binding energy of the cluster to decrease, and the rate of escape of stars through evaporation to increase (Chernoff, Kochanek, & Shapiro 1986). In addition, in some cases, "shock-induced relaxation" can be almost as important as two-body relaxation in the overall evolution of the cluster (Gnedin et al. 1999; Gnedin & Ostriker 1997). Both the energy shift and the relaxation induced by tidal shocking can be incorporated in our Monte Carlo method by assuming an orbit for the cluster around the Galactic center and introducing an appropriate perturbation to the energy of the stars each time the cluster passes through the disk. This can be done without adding much computational overhead to the problem, since tidal shocking only occurs twice during the orbital period of the cluster. The ability of the Monte Carlo method to model such effects simultaneously with a realistic treatment of the internal dynamical evolution of the cluster makes it a very useful tool in verifying and extending previous results obtained using other methods.

The star-by-star representation of the system in Monte Carlo simulations makes it easy to study of the evolution of a particular population of stars within a cluster. For example, the evolution of a population of neutron stars could be followed closely, to help predict their properties and expected distributions within clusters. Of particular interest are M15 and 47 Tuc, which have both been the targets of several highly successful searches for pulsars (Anderson 1992; Robinson et al. 1995; Camilo et al. 2000). The observed properties of pulsars in these clusters are found to be very different. The pulsars in 47 Tuc are all millisecond pulsars, and most are in short-period binaries, while those in M15 are mostly single recycled pulsars with longer pulse periods. This suggests that these two clusters may provide very different dynamical environments for the formation of recycled pulsars.

2. THE MONTE CARLO METHOD

2.1. *Overview*

Our basic algorithm for doing stellar dynamics is based on the "orbit-averaged Monte Carlo method" developed by Hénon (1971a, 1971b). The method was later used and improved by Stodółkiewicz (1982, 1985, 1986). It has also recently been used by Spurzem & Giersz (1996) to follow the evolution of hard three-body binaries in a cluster with equal point-mass stars. New results using Stodółkiewicz's version of the method were also presented recently by Giersz (1998). In earlier implementations of the Monte Carlo method with $N \sim 10^3$, each particle in the simulation was a "superstar," representing many individual stars with similar orbital properties. In our implementation, with $N \sim 10^5$–10^6, we treat each particle in the simulation as a single star. We have also modified Hénon's original algorithm to allow the time step to be made much smaller in order to resolve the dynamics in the core more accurately.

In the simplest case of a spherical system containing N point masses the algorithm can be summarized as follows. We begin by assigning to each star a mass, radius and velocity by sampling from a spherical and isotropic distribution function (for example, the Plummer model). Once the positions and masses of all stars are known, the gravitational potential of the cluster is computed assuming spherical symmetry. The energy and angular momentum of each star are then calculated. Energy and angular momentum are perturbed at each time step to simulate the effects of two-body and three-body relaxation. The perturbations depend on each star's position and velocity, and on the density of stars in its neighborhood. The time step should be a fraction of the relaxation time for the cluster (which is larger than the dynamical time by a factor $\propto N/\ln N$). The perturbation of the energy and angular momentum of a star at each time step therefore represents the cumulative effect of many small (and distant) encounters with other stars. Under the assumption of spherical symmetry, the cross sections for these perturbations can be computed analytically. The local number density is computed using a sampling procedure. Once a new energy and angular momentum is assigned to each star, a new realization of the system is generated by assigning to each star a new position and velocity in an orbit that is consistent with its new energy and angular momentum. In selecting a new position for each star along its orbit, each position is weighted by the amount of time the star spends around that position. Using the new positions, the gravitational potential is then recomputed for the entire cluster. This procedure is then repeated over many time steps. After every time step, all stars with positive total energy (see § 2.7) are removed from the computation since they are no longer bound to the cluster and are hence considered lost from the cluster instantly on the relaxation timescale. The method allows stars to have arbitrary masses and makes it very easy to allow for a stellar mass spectrum in the calculations.

We now describe our implementation of the Monte Carlo method in detail. For completeness, we also include some of the basic equations of the method. For derivations of these equations, and a more detailed discussion of the basic method, see Hénon (1971b), Stodółkiewicz (1982), and Spitzer (1987).

2.2. *Initial Conditions*

The initial model is assumed to be in dynamical equilibrium, so that the potential does not change on the crossing timescale. This is important since the Monte Carlo method uses a time step that is of the order of the relaxation time, and hence cannot handle the initial phase of "violent relaxation" during which the potential changes on the dynamical timescale. Under the assumption of spherical symmetry, the distribution function for such an equilibrium system can be written in the form $f = \Psi(E, J)$, where E and J are the energy per unit mass, and angular momentum per unit mass,

$$E = \Phi(r) + \tfrac{1}{2}(v_r^2 + v_t^2) , \qquad (1)$$

$$J = r v_t . \qquad (2)$$

Here r is the distance from the cluster center, v_r is the radial velocity, v_t is the transverse velocity, and $\Phi(r)$ is the gravitational potential. In principle, the initial distribution function $\Psi(E, J)$ can be arbitrary. However, in practice, computing a self-consistent potential for an arbitrary distribution function can be quite difficult. Since the method requires the initial potential $\Phi(r)$ to be known, a simple initial model is usually selected so as to allow the potential to be computed quasianalytically. Common examples are the sequence of King models and the Plummer model.

Once the number of stars N is selected, the initial condition is constructed by assigning to each star values for r, v_r, v_t, and m, consistent with the selected model. Once the positions and masses of all the stars are known, the gravitational potential Φ is computed as a function of distance from the center. The energy per unit mass E, and angular momentum per unit mass J of each star are then computed using equations (1) and (2).

2.3. *The Gravitational Potential*

We compute the mean potential of the cluster by summing the potential due to each star, under the assumption of spherical symmetry. We use only the radial position r of each star (since we assume spherical symmetry, we can neglect the angular positions of the stars, to a very good approximation). We begin by sorting all the stars by increasing radius. Then the potential at a point r, which lies between two stars at positions r_k and r_{k+1}, is given by

$$\Phi(r) = G\left(-\frac{1}{r} \sum_{i=1}^{k} m_i - \sum_{i=k+1}^{N} \frac{m_i}{r_i} \right) . \qquad (3)$$

For any two neighboring stars at distances r_k and r_{k+1}, the mass contained within the radius r remains constant for $r_k < r < r_{k+1}$. Hence, we can compute the potential at r, if the potentials $\Phi_k = \Phi(r_k)$ and $\Phi_{k+1} = \Phi(r_{k+1})$ are known, as

$$\Phi(r) = \Phi_k + \left(\frac{1/r_k - 1/r}{1/r_k - 1/r_{k+1}} \right) (\Phi_{k+1} - \Phi_k) . \qquad (4)$$

At each time step, we store precomputed values of $\Phi_k = \Phi(r_k)$, for each star k in the cluster. The potential at an arbitrary point r can then be quickly computed simply by finding the index k such that $r_k \le r \le r_{k+1}$ and then using equation (4).

We now describe the process of evolving the system through one complete time step.

2.4. *Two-Body Relaxation and Time step Selection*

We simulate the effect of interactions during each time step Δt by perturbing the energy and angular momentum of each star in the cluster. The perturbations ΔE and ΔJ for a star are determined by computing a single *effective* encounter between the star and its nearest neighbor (in terms of distance from the center, since we assume spherical symmetry). During such an encounter, the two stars exchange kinetic energy, but the total energy is conserved. In the center of mass frame of the two interacting stars, the magnitude of the velocity does not change; instead the velocity is deflected through an angle β.

In the original method described by Hénon (1971b), the time step used was a small fraction of the relaxation time for *the entire cluster*. Although the time step computed in this way is suitable for the outer regions of the cluster, it is too large to provide an accurate representation of the relaxation in the core, especially in the later stages of cluster evolution, where the relaxation time in the core can be many orders of magnitude smaller than in the outer regions. This caused the inner regions of the cluster to be underrelaxed. The limited computational resources available at that time did not permit the time step to be made much smaller without slowing down the computation to a crawl. The greatly increased computational power available today allows us to use a time step that is small enough to resolve the relaxation process in the core, even for systems with $N \gtrsim 10^5$.

To provide an accurate description of the overall relaxation of the cluster, each effective encounter should give the correct mean value of the change in energy at each position. We achieve this by selecting the effective deflection angle β_e for the encounter (in the center of mass frame of the two interacting stars) as follows. If the masses of the two stars are m_1 and m_2, and their velocities v_1 and v_2, respectively, then the kinetic energy changes can be written as

$$\Delta KE_1 = m_1 v_1 \Delta v_1 + \tfrac{1}{2} m_1 (\Delta v_1)^2 \,, \qquad (5)$$

$$\Delta KE_2 = m_2 v_2 \Delta v_2 + \tfrac{1}{2} m_2 (\Delta v_2)^2 \,, \qquad (6)$$

where Δv_1 and Δv_2 are the changes in the velocities during the encounter. Since the total kinetic energy in each encounter is conserved, the mean value of the first terms on the right-hand side of equations (5) and (6) must equal the mean value of the second terms (with the opposite sign). This indicates that in order to get a good representation of the energy exchange between stars in the relaxation process, we must consider the mean value of $m_1(\Delta v_1)^2$ during each time step.

The change in velocity Δv_1 during an encounter with a deflection angle β, can be calculated from elementary mechanics as (see, e.g., Spitzer 1987, eq. [2]–[6]),

$$(\Delta v_1)^2 = 4 \frac{m_2^2}{(m_1 + m_2)^2} w^2 \sin^2(\beta/2) \,, \qquad (7)$$

where w is the relative speed of the two stars before the encounter. The mean overall rate of change in the velocity $\langle(\Delta v_1)^2\rangle$ due to many distant (weak) encounters of the star with other cluster stars can then be calculated by averaging over the impact parameter (see Spitzer 1987, eq. [2]–[8]). Using this, the mean change in the velocity in the time Δt is given by

$$\langle(\Delta v_1)^2\rangle = 8\pi G^2 \nu \Delta t \langle m_2^2 w^{-1}\rangle \ln \Lambda \,, \qquad (8)$$

where $\ln \Lambda \equiv \ln(\gamma N)$ is the Coulomb logarithm (γ is a constant ~ 0.1; see § 3.1), and ν is the local number density of stars. We obtain the correct mean value of $m_1(\Delta v_1)^2$ by equating the right-hand side of equations (7) and (8), giving

$$\left\langle 4 \frac{m_1 m_2^2}{(m_1 + m_2)^2} w^2 \sin^2(\beta/2) \right\rangle = 8\pi G^2 \nu \Delta t \langle m_1 m_2^2 w^{-1}\rangle \ln(\gamma N) \,. \qquad (9)$$

Equation (9) relates the time step Δt to the deflection angle β for the encounter. Thus, in order to get the correct mean value of $m_1(\Delta v_1)^2$ for the star during the time Δt, we can define the *effective* deflection angle β_e for the representative encounter, as

$$\sin^2(\beta_e/2) = 2\pi G^2 \frac{(m_1 + m_2)^2}{w^3} \nu \Delta t \ln(\gamma N) \,. \qquad (10)$$

In addition to using the correct mean value of $m_1(\Delta v_1)^2$, we can also require that its variance be correct. To compute the variance, we must calculate the mean value of $(\Delta v_1)^4$. Using equation (7), we have

$$(\Delta v_1)^4 = 16 \frac{m_2^4}{(m_1 + m_2)^4} w^4 \sin^4(\beta/2) \,. \qquad (11)$$

We then use Spitzer's equations (2)–(5), and again integrate over the impact parameter to get the mean value of $(\Delta v_1)^4$ in the time Δt,

$$\langle(\Delta v_1)^4\rangle = 16\pi G^2 \frac{m_2^4}{(m_1 + m_2)^2} w \nu \Delta t \,. \qquad (12)$$

Comparing equations (11) and (12), we see that, in order to have the correct variance of $m_1(\Delta v_1)^2$, we should have

$$\sin^4(\beta_e/2) = \pi G^2 \frac{(m_1 + m_2)^2}{w^3} \nu \Delta t \,. \qquad (13)$$

Consistency between equations (10) and (13) gives the relation between the number of stars in the system, and the effective deflection angle that must be used,

$$\sin^2(\beta_e/2) = \frac{1}{2 \ln(\gamma N)} \,. \qquad (14)$$

This relation indicates that for large N, the effective deflection angle must be small, while as N decreases, close encounters become more important. If the time step is too large, then $\langle \sin^2(\beta/2)\rangle$ is also too large, and the system is underrelaxed. Hence the time step used should be sufficiently small so as to get a good representation of the relaxation process in the cluster. In addition, the local relaxation time varies greatly with distance from the cluster center. In practice we use the shortest relaxation time in the core to compute the time step. We first evaluate the local density ρ_c in the core and the approximate core radius $r_c = (3v_c^2/4\pi G\rho_c)^{1/2}$. We then compute the time step Δt using equation (10) and requiring that the average value of $\sin^2(\beta_e/2)$ for the stars within the core radius r_c be sufficiently small. The value of $\sin^2(\beta_e/2)$ given by equation (14) varies only slightly between 0.046 and 0.072 for N between 10^4 and 5×10^5 (assuming $\gamma \simeq 0.1$). Hence for all our simulations, we require that $\sin^2(\beta_e/2) \lesssim 0.05$.

Equation (10) is then used to compute the effective deflection angle for all stars in the cluster. The local number density ν is computed by averaging over the nearest p stars. We find that using a value of p between 20 and 50 gives the best results for $N \sim 10^5$. We find that the difference in the core-collapse times obtained for various test models using values of p between 20 and 50 is less than 1%. Of course, the value of p should not be too large so as to maintain a truly local estimate of the number density. We use a value of $p = 40$ in all our calculations, which gives consistently good agreement with published results.

2.5. *Computing the Perturbations ΔE and ΔJ during an Encounter*

To compute the velocity perturbation during each time step, a single representative encounter is computed for each star, with its nearest neighbor in radius. Selecting the nearest neighbor ensures that the correct local velocity distribution is sampled and also accounts for any anisotropy in the orbits. Because of spherical symmetry, selecting the nearest neighbor in radius is equivalent to selecting the nearest neighbor in three dimensions, since only the velocity (and not the position) of the nearest neighbor is used in the encounter. Following Hénon's notation, we let (r, v_r, v_t) and (r', v_r', v_t') represent the phase space coordinates of the two interacting stars, with masses m and m', respectively. In addition to these parameters, the angle ψ of the plane of relative motion defined by $(\boldsymbol{r}' - \boldsymbol{r}, \boldsymbol{v}' - \boldsymbol{v})$ with some reference plane is selected randomly between 0 and 2π, since the distribution of field stars is assumed to be spherically symmetric.

We take our frame of reference such that the z-axis is parallel to $\boldsymbol{r}$, and the (x, z)-plane contains $\boldsymbol{v}$. Then the velocities of the two stars are given by

$$\boldsymbol{v} = (v_t, 0, v_r) , \quad \boldsymbol{v}' = (v_t' \cos \phi, v_t' \sin \phi, v_r') , \qquad (15)$$

where ϕ is also randomly selected between 0 and 2π, since the transverse velocities are isotropic because of spherical symmetry. The relative velocity $\boldsymbol{w} = (w_x, w_y, w_z)$ is then

$$\boldsymbol{w} = (v_t' \cos \phi - v_t, v_t' \sin \phi, v_r' - v_r) . \qquad (16)$$

We now define two vectors $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ with the same magnitude as $\boldsymbol{w}$, such that $\boldsymbol{w}_1$, $\boldsymbol{w}_2$, and $\boldsymbol{w}$ are mutually orthogonal. The vectors $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ are given by

$$\boldsymbol{w}_1 = (w_y w/w_p, -w_x w/w_p, 0) , \qquad (17)$$

$$\boldsymbol{w}_2 = (-w_x w_z/w_p, -w_y w_z/w_p, w_p) , \qquad (18)$$

where $w_p = (w_x^2 + w_y^2)^{1/2}$. The angle ψ is measured from the plane containing the vectors $\boldsymbol{w}$ and $\boldsymbol{w}_1$. The relative velocity of the two stars after the encounter is given by

$$\boldsymbol{w}^* = \boldsymbol{w} \cos \beta + \boldsymbol{w}_1 \sin \beta \cos \psi + \boldsymbol{w}_2 \sin \beta \sin \psi , \qquad (19)$$

where β is the deflection angle computed in § 2.4. The new velocities of the two stars after the interaction are then given by

$$\boldsymbol{v}^* = \boldsymbol{v} - \frac{m'}{m + m'} (\boldsymbol{w}^* - \boldsymbol{w}) , \qquad (20)$$

$$\boldsymbol{v}'^* = \boldsymbol{v}' + \frac{m}{m + m'} (\boldsymbol{w}^* - \boldsymbol{w}) . \qquad (21)$$

The new radial and transverse velocities for the first star are given by $v_r^* = v_z^*$, and $v_t^* = (v_x^{*2} + v_y^{*2})^{1/2}$, from which we compute the new orbital energy E and angular momentum J as $E^* = \Phi(r) + \frac{1}{2}(v_r^{*2} + v_t^{*2})$, and $J^* = r v_t^*$. Similar quantities E'^* and J'^* are also computed for the second star.

2.6. *Computing New Positions and Velocities*

Once the orbits of all the stars are perturbed, i.e., new values of E and J are computed for each star, a new realization of the system is generated, by selecting a new position for each star in its new orbit, in such a way that each position in the orbit is weighted by the amount of time that the star spends at that position. To do this, we begin by computing the pericenter and apocenter distances, $r_{\rm min}$ and $r_{\rm max}$, for each star. The orbit of a star in the cluster potential is a rosette, with r oscillating between $r_{\rm min}$ and $r_{\rm max}$, which are roots of the equation

$$Q(r) = 2E - 2\Phi(r) - J^2/r^2 = 0 . \qquad (22)$$

See Binney & Tremaine (1987, § 3.1) for a general discussion, and see Hénon (1971b, eqs. [41]–[45]) for a convenient method of solution. The new position r should now be selected between $r_{\rm min}$ and $r_{\rm max}$, in such a way that the probability of finding r in an interval dr is equal to the fraction of time spent by the star in the interval during one orbit, i.e.,

$$\frac{dt}{P} = \frac{dr/|v_r|}{\int_{r_{\rm min}}^{r_{\rm max}} dr/|v_r|} , \qquad (23)$$

where P is the orbital period, and $|v_r|$ is given by

$$|v_r| = [2E - 2\Phi(r) - J^2/r^2]^{1/2} = [Q(r)]^{1/2} . \qquad (24)$$

Thus the value of r should be selected from a probability distribution that is proportional to $f(r) = 1/|v_r|$. Unfortunately, at the pericenter and apocenter points ($r_{\rm min}$ and $r_{\rm max}$), the radial velocity v_r is zero, and the probability distribution becomes infinite. To overcome this problem, we make a change of coordinates by defining a suitable function $r = r(s)$ and selecting a value of s from the distribution

$$g(s) \equiv \frac{1}{|v_r|} \frac{dr}{ds} . \qquad (25)$$

We must select the function $r(s)$ such that $g(s)$ remains finite in the entire interval. A convenient function $r(s)$ that satisfies these requirements is given by

$$r = \tfrac{1}{2}(r_{\rm min} + r_{\rm max}) + \tfrac{1}{4}(r_{\rm max} - r_{\rm min})(3s - s^3) , \qquad (26)$$

where s lies in the interval -1 to 1. We then generate a value for s, which is consistent with the distribution $g(s)$, using the von Neumann rejection technique. Equation (26) then gives a corresponding value for r that is consistent with the distribution function $f(r)$.

The magnitude of the new radial velocity v_r is computed using equation (24), and its sign is selected randomly. The transverse velocity is given by $v_t = J/r$.

Once a new position is selected for each star using the above procedure, the gravitational potential $\Phi(r)$ is recomputed as described in § 2.3. This completes the time step, and allows the next time step to be started.

Note that the gravitational potential used to compute new positions and velocities of the stars is from the previous time step. The new potential can only be computed *after* the new positions are assigned, and it is then used to recompute the positions in the next time step. Thus the computed potential always lags slightly behind the actual potential of the system. The exact potential is known only at the initial condition. This only introduces a small systematic error in the computation, since the potential changes significantly only on the relaxation timescale.

A more important source of error, especially in computing the new energies of the stars after the potential is recomputed, is the random fluctuation of the potential in the core, which contains relatively few stars, but has a high number density. Since the derivative of the potential is also steepest in the core, a small error in computing a star's position in the core can lead to a large error in computing its energy. As the simulation progresses, this causes a slow but consistent leak in the total system energy. The magnitude of this error (i.e., the amount of energy lost per time step) depends partly on the number of stars N in the system. For large N, the grid on which the potential is precomputed (see § 2.3) is finer, and the number of stars in the core is larger, which reduces the noise in the potential. The overall error in energy during the course of an entire simulation is typically of order a few percent for $N = 10^5$ stars. In any realistic simulation, the actual energy gain or loss due to real physical processes such as stellar evolution, escape of stars through a tidal boundary, and interactions involving binaries, is at least an order of magnitude greater than this error. Hence we choose not to renormalize the energy of the system, or employ any other method to artificially conserve the energy of the system, which could affect other aspects of the evolution.

Another possible source of error in Monte Carlo simulations, which was noted by Hénon (1971b) is the "spurious relaxation" effect. This is the tendency for the system to relax because of the potential fluctuations from one time step to the next, even in the absence of orbital perturbations due to two-body relaxation. However, this effect is significant only for simulations done with very low $N \sim 10^2$–10^3. In test calculations performed with $N \sim 10^4$–10^5 and two-body relaxation explicitly turned off (by setting the scattering angle $\beta_e = 0$ in eq. [10]), we find no evidence of spurious relaxation. Indeed Hénon (1971b) himself showed that spurious relaxation was not significant in his models for $N \gtrsim 10^3$.

2.7. *Escape of Stars and the Effect of a Tidal Boundary*

For an isolated system, the gradual evaporation of stars from the cluster is computed in the following way. During each time step, after the perturbations ΔE and ΔJ are computed, all stars with a positive total energy (given by eq. [1]) are assumed to leave the cluster on the crossing timescale. They are therefore considered lost immediately on the relaxation timescale, and removed from the simulation. The mass of the cluster (and its total energy) decreases gradually as a result of this evaporation process.

As a simple first step to take in to account the tidal field of the Galaxy, we include an effective tidal boundary around the cluster, at a distance $r_t \simeq R_g(M_{\rm cluster}/3M_g)^{1/3}$, where R_g is the distance of the cluster from the Galactic center and M_g is the mass of the Galaxy (approximated as a point mass). The tidal radius is roughly the size of the Roche lobe of the cluster in the field of the Galaxy. Once the initial tidal radius r_{t0} is specified, the tidal radius at a subsequent time t during the simulation can be computed by $r_t(t) = r_{t0}[M_{\rm cluster}(t)/M_{\rm cluster}(0)]^{1/3}$. After each time step, we remove all stars with an apocenter distance $r_{\rm max}$ greater than the tidal radius, since they are lost from the cluster on the crossing timescale. As the cluster loses stars as a result of evaporation and the presence of the tidal boundary, its mass decreases, which causes the tidal boundary to shrink, in turn causing even more stars to be lost. The total mass loss due to a tidal boundary can be very significant, causing up to 90% of the mass to be lost (depending on the initial model) over the course of the simulation (see § 3.2).

2.8. *Units*

Following the convention of most previous studies, we define dynamical units so that $[G] = [M_0] = [-4E_0] = 1$, where M_0 and E_0 are the initial total mass and total energy of the system (Hénon 1971b). Then the units of length L, and time T are given by

$$L = GM_0^2(-4E_0)^{-1} , \quad \text{and} \quad T = GM_0^{5/2}(-4E_0)^{-3/2} . \tag{27}$$

We see that L is basically the virial radius of the cluster, and T is of the order of the initial dynamical (crossing) time. To compute the evolution of the cluster on a relaxation timescale, we rescale the unit of time to $TN_0/\ln(\gamma N_0)$, which is of the order of the initial relaxation time. Using this unit of time allows us to eliminate the $\ln(\gamma N)$ dependence of the evolution equations. The only equation that explicitly contains the evolution time is equation (10), which relates the time step and the effective deflection angle. In our units, equation (10) can be written as,

$$[\sin^2(\beta_e/2)] = 2\pi \frac{([m_1] + [m_2])^2}{[w]^3} [v][\Delta t]N , \tag{28}$$

where $[q]$ indicates a quantity q expressed in our simulation units. Using a unit of time that is proportional to the initial relaxation time has the advantage that the evolution timescale is roughly independent of the number of stars N once an initial model has been selected. This is only true approximately, for isolated systems of equal-mass stars, with no other processes that depend explicitly on the number of stars (such as stellar evolution or mass segregation). For example, the half-mass relaxation time for the Plummer model,

$$t_{\rm rh} = \frac{0.138N}{\ln(\gamma N)} \left(\frac{r_h^3}{GM}\right)^{1/2} , \tag{29}$$

is always 0.093 in our units, independent of N.

The dynamical units defined above are identical to the standard N-body units (Heggie & Mathieu 1986). Hence to convert the evolution time from N-body time units to our Monte Carlo units, we must simply multiply by a factor $\ln(\gamma N_0)/N_0$.

2.9. *Numerical Implementation*

We have implemented our Monte Carlo code on the SGI/CRAY Origin2000 parallel supercomputer at the National Center for Supercomputing Applications (NCSA), and at Boston University. Our parallelized code can be used to get significant speedup of the simulations, using up to eight processors, especially for large N simulations. This ability to perform large N simulations will be particularly useful for doing realistic simulations of very large globular clusters such as 47 Tuc (with $N \gtrsim 10^6$ stars). A simulation with $N = 10^5$ stars can be completed in approximately 15–20 CPU hours on the Origin2000, which uses MIPS

R10000 processors. For comparison, a simulation of this size would take ~6 months to complete using the GRAPE-4, which is the fastest available hardware for N-body methods.

The most computationally intensive step in the simulation is the calculation of the new positions of stars. The operation involves solving for the roots of an equation (eq. [22]) using the indexed values of the positions of the N stars. We find that the most efficient method to solve for the roots in this case is the simple bisection method (e.g., Press et al. 1992), which requires $\sim N \log_2 N$ steps to converge to the root. Hence the computation of the positions and velocities also scales as $\sim N \log_2 N$ in our method. The next most expensive operation is the evaluation of the potential at a given point r. As described in § 2.3, this requires finding k such that $r_k \leq r \leq r_{k+1}$ and then using equation (4). This search can again be done easily using the bisection algorithm. However, since the evaluation of the potential is required several times for each star, in each time step, it is useful to tabulate the values of k on fine grid in r at the beginning of the time step. This allows the required values of k to be found very quickly, at the minor cost of using more memory to store the table. The rest of the steps in the simulation scale almost linearly with N. This makes the overall computation timescale (theoretically) as $N \log_2 N$.

In Figure 1, we show the scaling of the wall-clock time with the number of processors, and also the scaling of the overall computation time with the number of stars N in the simulation. The overall computation time is consistent with the theoretical estimate for $N \lesssim 10^5$. For larger N, the computation time is significantly higher, because of the less efficient use of cache memory and other hardware inefficiencies that are introduced while handling large arrays. For N in the range 1–5 × 10^5, we find that the actual computation timescales as $\sim N^{1.4}$.

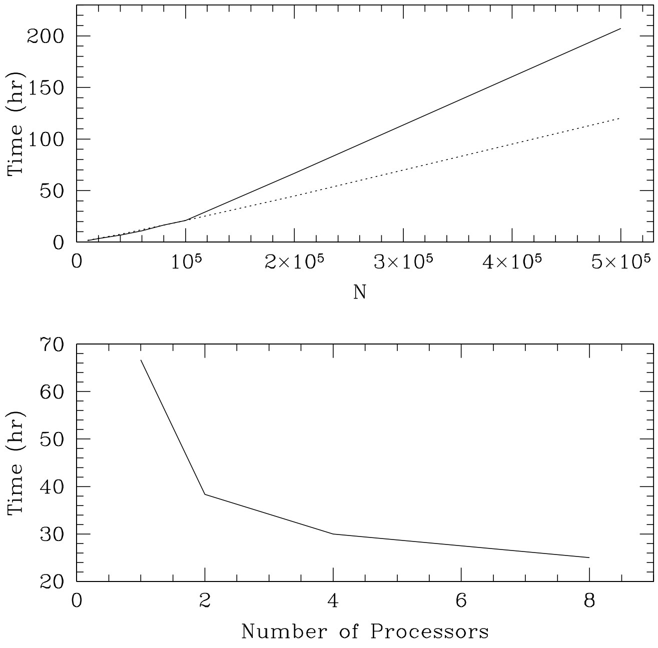

FIG. 1.—Top frame shows the total computation time required (for an initial Plummer model evolved up to core collapse) using one processor for simulations with up to $N = 5 \times 10^5$. The dotted line indicates the theoretically estimated scaling of the computation time as $\sim N \log_2 N$. In practice, we find that the computation timescales as $\sim N^{1.4}$ for N = 1–5 × 10^5. The bottom frame shows the scaling of the computation time ("wall-clock time") with the number of processors for $N = 2 \times 10^5$.

We find that we can easily reduce the overall computation time by a factor of ≈3 by using up to eight processors. The scaling is most efficient for two–four processors for simulations with $N \sim$ 1–5 × 10^5. The scaling gets progressively worse for more than eight processors. This is in part caused by the distributed shared-memory architecture of the Origin2000 supercomputer, which allows very fast communication between the nearest two–four processors, but slower communication between the nearest eight processors. Beyond eight processors, the communication is even slower, since the processors are located on different nodes. The most suitable architecture for implementing the parallel Monte Carlo code would be a truly shared memory supercomputer, with roughly uniform memory access times between processors. Our code is implemented using the Message Passing Interface (MPI) parallelization library, which is actively being developed and improved. The MPI standard is highly portable, and available on practically all parallel computing platforms in use today. The MPI library is optimized for each platform and automatically takes advantage of the memory architecture to the maximum extent possible. Hence we expect that future improvements in the communication speed and memory architectures will make our code scale even better. We are also in the process of improving the scaling of the code to a larger number of processors by designing a new algorithm for reducing the amount of communication required between processors. This will be described in detail in a subsequent paper, where we incorporate primordial binary interactions in our code.

3. TEST RESULTS

In this section, we describe our first results using the new Monte Carlo code to compute the evolution of the Plummer and King models. We explore the evolution of the Plummer model in detail, and compare our results with those obtained using Fokker-Planck and N-body methods. We also compare core-collapse times and mass-loss rates for the series of King models (W_0 = 1–12), including a tidal radius, with similar results obtained by Quinlan (1996) using a one-dimensional Fokker-Planck method.

3.1. *Evolution of an Isolated Plummer Model*

We first consider the evolution of a cluster with the Plummer model (which is a polytropic model, with index $n = 5$; see, e.g., Binney & Tremaine 1987) as the initial condition. Perhaps the best known result for single-component systems, is the expected homologous evolution of the halo, leading to the eventual development of a power-law density profile between the core and the outer halo, during the late phases of evolution. At late times the cluster evolves through a sequence of nearly self-similar configurations, with the core contracting and a power-law halo with density $\rho \propto r^{-\beta}$ expanding out. The development of this power law has been predicted theoretically (Lynden-Bell & Eggleton 1980; Heggie & Stevenson 1988), and verified using direct Fokker-Planck integrations (Cohn 1980). The exponent β is theoretically and numerically estimated to be about 2.2 (Spitzer 1987). However, since the theoretical derivations are based on an analysis of the Fokker-Planck equation, it is not surprising that the numerical Fokker-Planck integrations (which solve the same Fokker-Planck equation numerically) reproduce the theoretical exponent exactly. Because of limitations in computing accurate density profiles using a small number of stars, this result has not been confirmed independently using an N-body simulation.

Here, we explore numerically for the first time the development of this power law using an independent method. Some early results were obtained using previous versions of the Monte Carlo method, but with a small number of stars $N \sim 10^3$ (Duncan & Shapiro 1982). Although the Monte Carlo method can be thought of as just another way of solving the Fokker-Planck equation, there are significant differences between solving the equation in the continuous limit ($N \to \infty$), as in direct Fokker-Planck integrations, and by using a discrete system with a finite N as in our method. There are also many subtle differences in the assumptions and approximations made in the two methods, and even in different implementations of the same method.

In Figures 2*a*–2*c* we show the density profile of the cluster at three different times during its evolution, up to core collapse. We start with an $N = 10^5$ isolated Plummer model, and follow the evolution up to core-collapse, which occurs at $t = t_{\rm cc} \simeq 15.2 t_{\rm rh}$. This simulation, performed with $N = 10^5$ stars, took about 18 CPU hours on the SGI/Cray Origin2000. In our calculations, the core-collapse time is taken as the time when the innermost Lagrange radius (radius containing 0.3% of the total mass of the cluster) becomes smaller than 0.001 (in our units described in § 2.8), at which point the simulation is terminated. Given the very rapid evolution of the core near core collapse, we find that we can determine the core-collapse time to within $\lesssim 1\%$. The accuracy is limited mainly by noise in the core. The value we obtain for $t_{\rm cc}/t_{\rm rh}$ is in very good agreement with other core-collapse times between 15–16$t_{\rm rh}$ for the Plummer model, reported using other methods. For example Quinlan (1996) obtains a core collapse time of $15.4 t_{\rm rh}$ for the Plummer model using a one-dimensional Fokker-Planck method, and Takahashi (1993) finds a value of $15.6 t_{\rm rh}$, using a variational method to solve the one-dimensional Fokker-Planck equation.

Figure 2*a* shows the density profile at an intermediate time $t = 11.4 t_{\rm rh}$ during the evolution. The dotted line indicates the initial Plummer profile. At this point in the evolution, we still see a well-defined core, with the core density increased by a factor of ~ 30 compared to the initial core density. We see the power-law density profile developing, with the best-fit index $\beta = 2.8$. In Figure 2*b*, we show the density profile just before core collapse, at $t = 15 t_{\rm rh}$. We see that the core density has now increased by a factor of $\sim 10^4$ over the initial core density. The power law is now clearly visible, with the best-fit index $\beta = 2.3$. Finally, in Figure 2*c*, we show the density profile at core-collapse, $t = 15.2 t_{\rm rh}$. The dashed line now indicates the *theoretical* power law with $\beta = 2.2$. We see that the actual density profile seems to approach the theoretical profile asymptotically as the system approaches core collapse. At this point in the evolution, the core density as measured in our simulation is about 10^6 times greater than the initial density. In a globular cluster with $N = 2 \times 10^5$, an average stellar mass $\langle m \rangle = 0.5\ M_\odot$, and a mean velocity dispersion $\langle v^2 \rangle^{1/2} = 5$ km s^{-1}, this would correspond to a number density of $\sim 2 \times 10^9$ pc^{-3}. Note that a real globular cluster is not expected to reach such high core densities, since the formation of binaries and the subsequent heating of the core due to binary interactions become significant at much lower densities. Numerical noise due to the extremely small size of the core makes it difficult to determine the core radius and density accurately at this stage. This also causes the numerical accuracy of the Monte Carlo method to deteriorate, forcing us to stop the computation. Thus, we find that the power-law structure of the density profile as the cluster approaches core collapse is consistent with theoretical predictions, and the power-law index approaches its theoretical value asymptotically during the late stages of core collapse.

Next we look at the evolution of the Lagrange radii (radii containing constant fractions of the total mass), and we compare our results with those of an equivalent N-body simulation. In Figure 3, we show the evolution of the Lagrange radii for an $N = 16384$ direct N-body integration by Makino (1996) and for our Monte Carlo integration with $N = 10^5$ stars. Time in the direct N-body integration is scaled to the initial relaxation time (the standard time unit in our Monte Carlo method) using equation (27) with $\gamma = 0.11$ (see Heggie & Mathieu 1986; Giersz & Heggie 1994; Makino 1996). The agreement between the N-body and Monte Carlo results is excellent over the entire range of Lagrange radii and time. The small discrepancy in the outer Lagrange radii is caused in part by a different treatment of escaping stars in the two models. In the Monte Carlo model, escaping stars are removed from the simulation and therefore not included in the determination of the Lagrange radii, whereas in the N-body model escaping stars are not removed. The difference is further explained by the effect of strong encounters, which is greater in the N-body simulation by a factor $\sim \ln(10^5)/\ln(16384)$, or about 20%. In an isolated cluster, the overall evaporation rate is very low (less than 1% of stars escape up to core collapse). In this regime, the escape of stars is dominated by strong interactions in the core. Since the orbit-averaged Fokker-Planck equation is only valid when the fractional energy change per orbit is small, it does not account for strong interactions. Hence, our Monte Carlo simulations cannot accurately predict the rate of evaporation from an isolated cluster (see, e.g., Binney & Tremaine 1987, § 8.4). This problem does not occur in tidally truncated clusters, where the escape rate is much higher, and is dominated by the diffusion of stars across the tidal boundary, and not by strong interactions.

In Figure 4 we show the evolution of various global quantities for the system during the same simulation as in Figure 3. The virial ratio ($K/|W|$, where K and W are the total kinetic and potential energies of the cluster) remains very close to 0.5 (within 1%), indicating that dynamical equilibrium is maintained very well during the entire simulation. The virial ratio provides a very good measure of the quality of our numerical results, since it is not controlled in our calculations (except for the initial model, which is constructed to be in equilibrium). We see that in the absence of a tidal radius, there is very little mass loss (less than 1%), and hence very little energy is carried away by escaping stars.

3.2. *Evolution of Isolated and Tidally Truncated King Models*

King models (King 1966) have long been used to fit observed profiles of globular clusters. They usually provide a very good fit for most clusters, except for those that have reached core collapse. A King model has a well-defined, nearly constant-density core, and a "lowered Maxwellian"

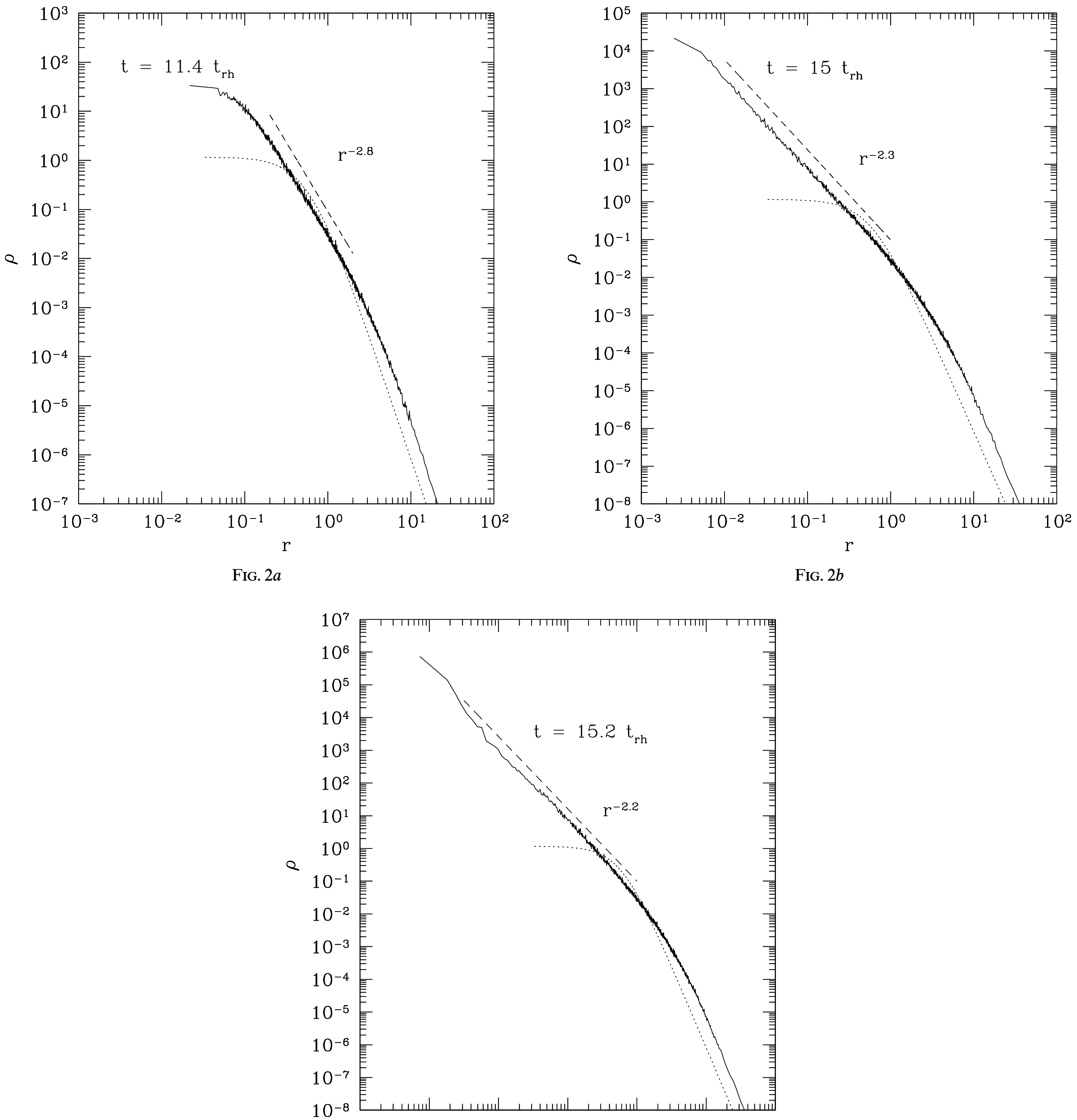

FIG. 2*a*

FIG. 2*b*

FIG. 2*c*

FIG. 2.—(*a*) Density profile at an intermediate time, $t = 11.4t_{\rm rh}$ during the evolution of an isolated Plummer model with $N = 10^5$ stars. The expected power-law in the density profile is clearly seen, with the best-fit exponent $\beta = 2.8$. The power-law exponent approaches its theoretical value of 2.2 as the cluster approaches core-collapse (see Figs. 2*b* & 2*c*). The dotted line indicates the initial Plummer profile. Units are defined in § 2.8. (*b*) Density profile at $t = 15t_{\rm rh}$ (just before core-collapse) for the same model as in Fig. 2*a*. The expected power-law in the density profile is now clearly seen, with the best-fit exponent $\beta = 2.3$, which is now closer to its theoretical value of 2.2. The core density is about 10^4 times greater than the initial density. (*c*) Density profile at $t_{\rm cc} = 15.2t_{\rm rh}$ (at core-collapse) for the same model as in Fig. 2*a*. The dashed line now indicates the *theoretical* power law, with exponent $\beta = 2.2$. The core density is almost 10^6 times greater than the initial density.

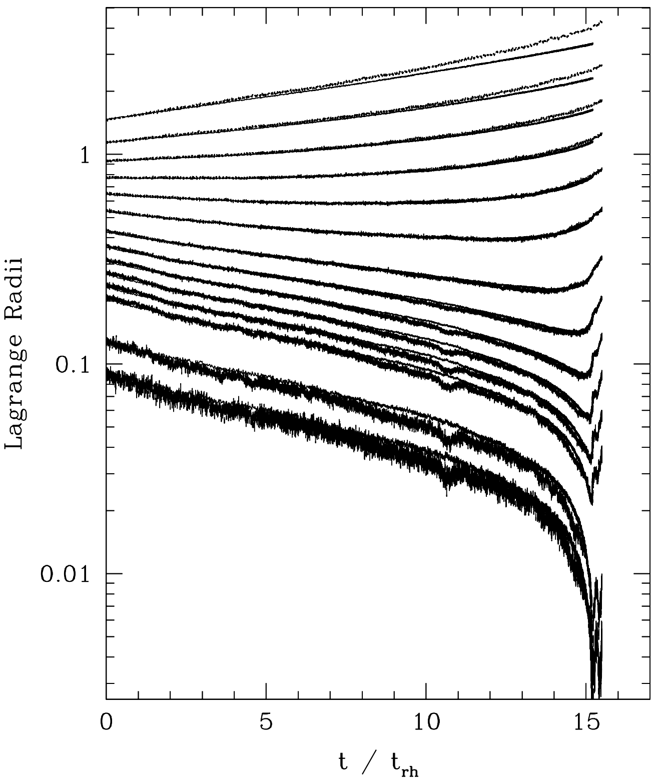

FIG. 3.—Lagrange radii indicating the evolution of the Plummer model, with $N = 10^5$ stars, compared with an N-body simulation with $N = 16{,}384$ stars. Lagrange radii shown correspond to radii containing 0.35, 1, 3.5, 5, 7, 10, 14, 20, 30, 40, 50, 60, 70, and 80% of the total mass. The Monte Carlo simulation is terminated at core-collapse, while the N-body simulation continues beyond core-collapse.

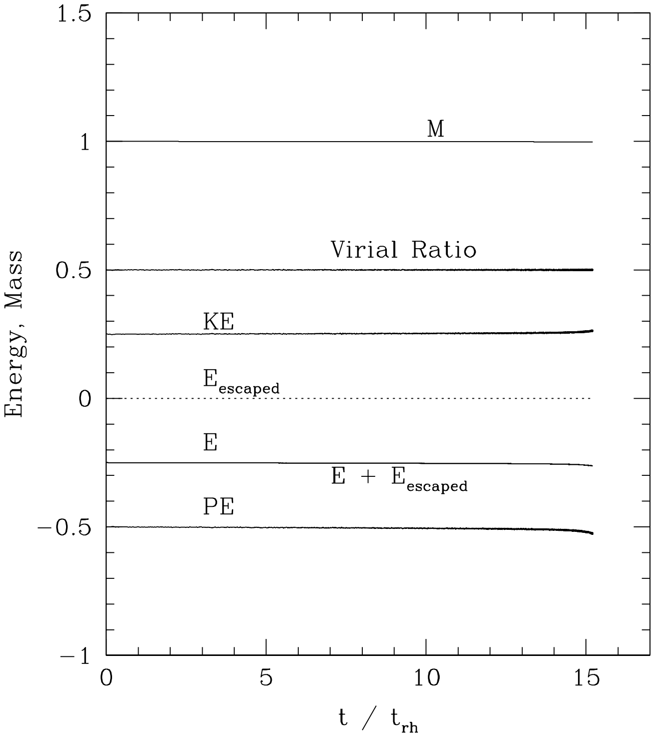

FIG. 4.—Evolution of the total mass and energies for the same Plummer model as in Fig. 3. The total mass loss at the time of core-collapse is 0.3%, and the total energy loss is about 4%. Most of the energy is lost during the late stages of evolution, with the energy loss up to $t = 10t_{\rm rh}$ being less than 1%. Here the energy carried away by escaping stars ($E_{\rm escaped}$) is negligible.

velocity distribution, which represents the presence of a finite tidal radius. A King model is usually specified in terms of the dimensionless central potential W_0 or, equivalently, the central concentration $c = \log (r_t/r_c)$, where r_t is the tidal radius, and r_c is the core radius.

We study the evolution of the entire family of King models from $W_0 = 1$ to $W_0 = 12$, in two different configurations. We first consider the evolution of an isolated cluster i.e., even though the initial King model is truncated at its finite tidal radius, we do not enforce that tidal boundary during the evolution, allowing the cluster to expand indefinitely. We compute the core-collapse times for the entire sequence of King models. We then redo the calculations with a tidal boundary in place, to determine the enhanced rate of mass loss from the cluster and the final remaining mass at the time of core collapse. We compare our results for the sequence of King models with equivalent results obtained by Quinlan (1996) using direct Fokker-Planck integrations in one dimension. In Table 1, we show the core collapse times for the various models, along with the equivalent results from Quinlan (1996). All our Monte Carlo calculations were performed using $N = 10^5$ stars. We see that the agreement in the core collapse times for isolated clusters is excellent (within a few percent for the low-W_0 models, and within 10% up to $W_0 = 9$). For $W_0 > 9$, the agreement is still good, considering that the models start off in a highly collapsed state and therefore have very short core-collapse times, which leads to larger fractional errors.

In Figure 5, we show the evolution of the Lagrange radii for a tidally truncated King model with $W_0 = 3$. The initial tidal radius is $\simeq 3.1$ times the virial radius. In this case, the mass loss through the tidal boundary is very significant, as is seen from the evolution of the outer Lagrange radii. The mass loss causes the tidal radius to constantly move inward, which further accelerates the process. Figure 6 shows the evolution of the total mass and energy of the tidally truncated cluster. Only 44% of the initial mass is retained in the cluster at core-collapse. Also, the binding energy of the cluster is significantly lower at core-collapse, since the escaping stars carry away mass as well as kinetic energy from the cluster. In contrast, the evolution of an isolated $W_0 = 3$ King model is very much like that of the isolated Plummer model described earlier, with a very low mass-loss rate, and a longer core-collapse time of $t_{\rm cc} = 17.7t_{\rm rh}$ (in excellent agreement with the value of $17.6t_{\rm rh}$ computed by Quinlan 1996).

Our results for clusters with a tidal boundary show systematic differences from the one-dimensional Fokker-Planck results of Quinlan (1996). We find that the mass loss through the tidal boundary is significantly higher for the low-concentration models ($W_0 < 6$) in the Fokker-Planck models. For the high-concentration ($W_0 > 6$) models, the difference between isolated models and tidally truncated models is small, and the agreement between the methods remains very good. Hence, for low W_0, our models undergo core collapse at a much later time compared to the Fokker-Planck models, and retain more mass at core collapse. This discrepancy is caused by the one-dimensional nature of the Fokker-Planck models. In one-dimensional Fokker-Planck calculations, stars are considered lost from the cluster when their energy is greater than the energy at the tidal radius. This clearly provides an overestimate of the escape rate, since it assumes the most extended radial orbits for stars, and ignores stars on more circular orbits with high angular

TABLE 1

CORE-COLLAPSE TIMES FOR KING MODELS

	ISOLATED t_{cc}/t_{rh}		TIDALLY TRUNCATED			
W_0	t_{cc}/t_{rh}	(Quinlan)	t_{cc}/t_{rh}	t_{cc}/t_{rh} (Quinlan)	(M_{final})	M_{final} (Quinlan)
1	18.1	17.89	10.0	5.98	0.30	0.10
2	17.9	17.85	10.8	7.74	0.37	0.17
3	17.7	17.61	12.0	9.49	0.44	0.24
4	17.3	17.24	12.9	11.26	0.53	0.33
5	15.9	16.37	13.3	12.73	0.64	0.44
6	13.9	14.49	12.4	12.94	0.76	0.57
7	10.6	10.84	9.30	10.50	0.86	0.72
8	5.32	5.79	5.21	5.76	0.88	0.85
9	2.10	2.25	2.01	2.25	0.96	0.92
10......	0.86	0.93	0.80	0.93	0.97	0.96
11......	0.41	0.47	0.40	0.47	0.99	0.98
12......	0.20	0.26	0.20	0.26	0.99	0.99

NOTE.—Core-collapse times for the sequence of isolated and tidally truncated King models, computed using $N = 10^5$ stars. Comparison is made with similar results obtained by Quinlan (1996) using a one-dimensional Fokker-Planck method.

momentum, which would have much smaller orbits at the same energy. In contrast, in the Monte Carlo method, the orbit of each star is computed using its energy *and* angular momentum, which allows the apocenter distance to be determined correctly. Stars are considered lost only if their apocenter distances from the cluster center are greater than the tidal radius. As stars on radial orbits are removed preferentially, this creates an anisotropy within the cluster, which affects the overall evolution. The artificially high rate of mass loss in one-dimensional Fokker-Planck simulations has also been pointed out recently in comparisons with N-body results (Portegies Zwart et al. 1998; Takahashi & Portegies Zwart 2000). These authors show that, with appropriate modifications, the results of two-dimensional

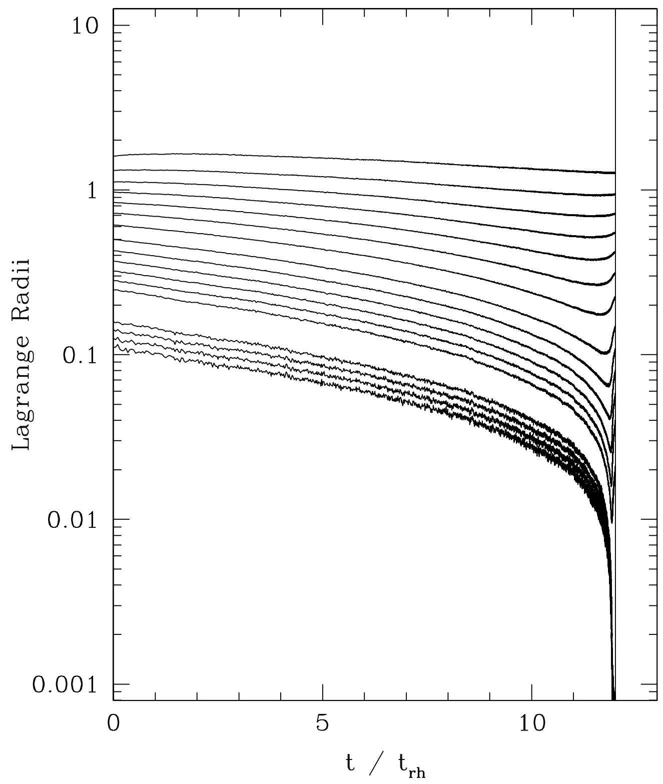

FIG. 5.—Lagrange radii for the evolution of a *tidally truncated* King model with $W_0 = 3$. The tidal boundary causes stars to be lost at a much higher rate compared to the isolated model. The vertical line indicates the core-collapse time $t_{cc} = 12.0t_{rh}$. The presence of the tidal boundary reduces the core-collapse time by about 32% compared to the isolated model. In contrast, the evolution of an *isolated* $W_0 = 3$ King model is very much like that of the Plummer model shown in Fig. 3, with a total mass loss less than 1%, and $t_{cc} = 17.7t_{rh}$

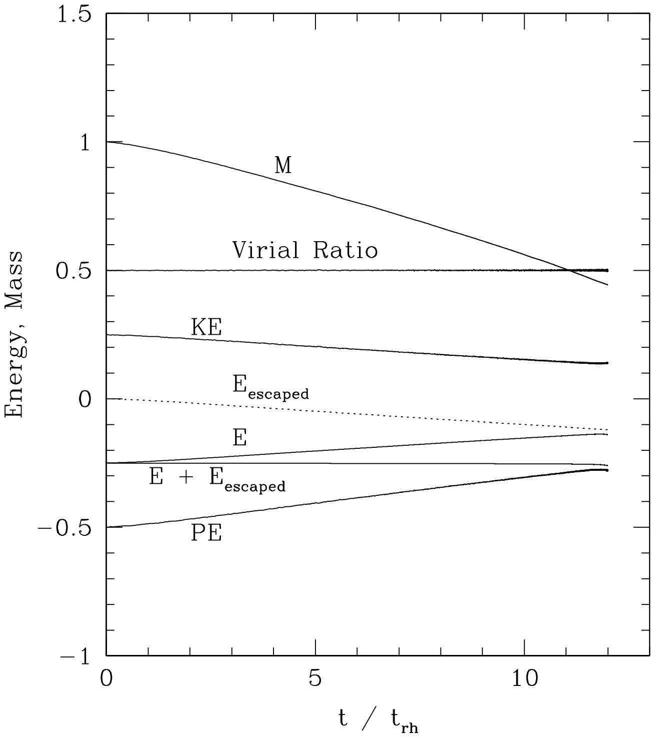

FIG. 6.—Evolution of the total mass and energies for the model shown in Fig. 5. Only 44% of the initial mass remains in the cluster at core collapse. The dotted line indicates the energy carried away by escaping stars. The large mass loss due to the tidal boundary causes the overall binding energy of the cluster to decrease significantly.

Fokker-Planck calculations can be made to agree much better with those from N-body simulations. Indeed, we find that our result for the $W_0 = 3$ model with a tidal boundary ($t_{cc} = 12.0t_{rh}$, and $M_{\rm final} = 0.44$) agrees much better with that obtained using the improved two-dimensional Fokker-Planck method, which gives $t_{cc} = 11.3t_{rh}$, and $M_{\rm final} = 0.34$ (K. Takahashi 1999, private communication). For further comparison, and to better understand the cause of the higher mass loss in the one-dimensional Fokker-Planck calculation, we have performed a Monte Carlo simulation using the same energy-based escape criterion that is used in the one-dimensional Fokker-Planck integrations. We find that using the energy-based escape criterion for $W_0 = 3$ gives $t_{cc} = 10.9t_{rh}$, and $M_{\rm final} = 0.30$, which agrees better with the one-dimensional Fokker-Planck result, but a significant discrepancy still remains. This is not surprising, since, even when using a one-dimensional escape criterion, our underlying method still remains two-dimensional. Again, our result agrees better with the corresponding result obtained by K. Takahashi (1999, private communication) using the energy-based escape criterion in his two-dimensional Fokker-Planck method, $t_{cc} = 10.2t_{rh}$, and $M_{\rm final} = 0.28$. It is reassuring to note that the differences between our two-dimensional results and one-dimensional Fokker-Planck results are also mirrored in the two-dimensional Fokker-Planck calculations of Takahashi. Since our Monte Carlo method is intrinsically two-dimensional, it is not possible for us to do a true one-dimensional (isotropic) calculation to compare results directly with one-dimensional Fokker-Planck calculations.

4. SUMMARY AND FUTURE DIRECTIONS

We have presented results obtained using our new Monte Carlo code for the evolution of clusters containing 10^5 stars, up to core collapse. We have compared our results with those of one-dimensional Fokker-Planck calculations (Quinlan 1996) for isolated as well as tidally truncated King models with $W_0 = 1$–12. We find very good agreement for the core-collapse times of isolated King models. For tidally truncated models (especially for $W_0 < 6$), we find that the escape rate of stars in our models is significantly lower than in the one-dimensional Fokker-Planck models. This is to be expected, since the one-dimensional Fokker-Planck models use an energy-based escape criterion, which does not account for the anisotropy in the orbits of stars, and hence overestimate the escape rate. This effect is most evident in tidally truncated clusters, since stars on radial orbits are preferentially removed, while those on more circular orbits (with the same energy) are not. In one case ($W_0 = 3$), we have verified that our results are in good agreement with those from new two-dimensional Fokker-Planck calculations (K. Takahashi 1999, private communication), which properly account for the velocity anisotropy, and use the same apocenter-based escape criterion as in our models. Further comparisons of our results with two-dimensional Fokker-Planck calculations will be presented in a subsequent paper (Joshi, Nave, & Rasio 2000). Our detailed comparison of the evolution of the Plummer model with an equivalent direct N-body simulation also shows excellent agreement between the two methods up to core collapse.

Our results clearly show that the Monte Carlo method provides a robust, scalable and flexible alternative for studying the evolution of globular clusters. Its strengths are complementary to those of other methods, especially N-body simulations, which are still prohibitively expensive for studying large systems with $N \gtrsim 10^5$. The Monte Carlo method requires more computational resources compared to Fokker-Planck methods, but it is several orders of magnitude faster than N-body simulations. The star-by-star representation of the system in this method makes it particularly well suited for studying the evolution of interesting subpopulations of stars within globular clusters, such as pulsars, blue stragglers, or black holes.

Our method also presents the interesting possibility of performing hybrid simulations that use the Monte Carlo method for the bulk of the evolution of a cluster up to the core collapse phase, and then switch to an N-body simulation to follow the complex core-collapse phase during which the high reliability of the N-body method is desirable. The discreteness of the Monte Carlo method, and the fact that it follows the same phase space parameters for a cluster as the N-body method, make it easy to switch from one method to the other during a single simulation.

In subsequent papers, we will present results for the dynamical evolution of clusters with different mass spectra, including the effects of mass loss due to stellar evolution. We are also in the process of incorporating primordial binaries in our Monte Carlo code, in order to follow the evolution in the post-core collapse phase. Dynamical interactions involving binaries will be treated using a combination of direct numerical integrations of the orbits on a case-by-case basis and precomputed cross sections. The cross sections will be obtained from separate sets of scattering experiments as well as fitting formulae (Sigurdsson & Phinney 1995; Heggie, Hut, & McMillan 1996, and references therein). Our code will also incorporate a simple treatment of stellar evolution in binaries, using an extensive set of approximate recipes and fitting formulae developed recently for StarLab (Portegies Zwart 1995). Simulations of clusters containing realistic numbers of stars *and* binaries will allow us for the first time ever to compute detailed predictions for the properties and distributions of all interaction products, including blue stragglers (from mergers of main-sequence stars), X-ray binaries and recycled pulsars (from interactions involving neutron stars) and cataclysmic variables (from interactions involving white dwarfs).

We thank Piet Hut and Stephen McMillan for many helpful discussions. We are grateful to Jun Makino and Koji Takahashi for kindly providing valuable data and answering numerous questions. This work was supported in part by NSF Grant AST-9618116 and NASA ATP Grant NAG 5-8460. F. A. R. was supported in part by an Alfred P. Sloan Research Fellowship. F. A. R. also thanks the Theory Division of the Harvard-Smithsonian Center for Astrophysics for hospitality. This work was also supported by the National Computational Science Alliance under Grant AST-970022N and utilized the SGI/Cray Origin2000 supercomputer at Boston University. NASA also supported this work through Hubble Fellowship grant HF-01112.01-98A awarded (to S. P. Z.) by the Space Telescope Science Institute, which is operated by the Association of Universities for Research in Astronomy, Inc., for NASA under contract NAS 5-26555. S. P. Z. is grateful to MIT for its hospitality and to Tokyo University for the extensive use of their GRAPE systems.

REFERENCES

Aarseth, S. 1998, in IAU Colloq. 172, Impact of Modern Dynamics in Astronomy, ed. J. Henrard & S. Ferraz-Mello (Dordrecht: Kluwer), 1
Anderson, S. B. 1992, Ph.D. thesis, California Institute of Technology
Bailyn, C. D. 1995, ARA&A, 33, 133
Binney, J., & Tremaine, S. 1987, Galactic Dynamics (Princeton: Princeton Univ. Press)
Breeden, J. L., Cohn, H. N., & Hut, P. 1994, ApJ, 421, 195
Camilo, F., Lorimer, D. R., Freire, P., Lyne, A. G., & Manchester, R. N. 2000, ApJ, 535, 975
Chang, J. S., & Cooper, G. 1970, J. Comput. Phys., 6, 1
Chernoff, D. F. 1996, in IAU Symp. 174, Dynamical Evolution of Star Clusters, ed. P. Hut & J. Makino (Dordrecht: Kluwer), 239
Chernoff, D. F., Kochanek, C. S., & Shapiro, S. L. 1986, ApJ, 309, 183
Chernoff, D. F., & Weinberg, M. D. 1990, ApJ, 351, 121
Cohn, H. 1979, ApJ, 234, 1036
———. 1980, ApJ, 242, 765
Davies, M. B., & Hansen, B. M. S. 1998, MNRAS, 301, 15
Di Stefano, R., & Rappaport, S. 1994, ApJ, 423, 274
Drukier, G. A., Cohn, H. N., Lugger, P. M., & Yong, H. 1999, ApJ, 518, 233
Duncan, M. J., & Shapiro, S. L. 1982, ApJ, 253, 921
Gao, B., Goodman, J., Cohn, H., & Murphy, B. 1991, ApJ, 370, 567
Giersz, M. 1998, MNRAS, 298, 1239
Giersz, M., & Heggie, D. C. 1994, MNRAS, 268, 257
Gnedin, O. Y., & Ostriker, J. P. 1997, ApJ, 474, 223
Gnedin, O. Y., Lee, H. M., & Ostriker, J. P. 1999, ApJ, 522, 935
Goodman, J., Heggie, D. C., & Hut, P. 1993, ApJ, 415, 715
Guhathakurta, P., Yanny, B., Schneider, D. P., & Bahcall, J. N. 1996, AJ, 111, 267
Heggie, D. C., Giersz, M., Spurzem, R., & Takahashi, K. 1998, in Highlights of Astronomy, Vol. 11, ed. J. Andersen, (Dordrecht; Kluwer), 591
Heggie, D. C., Hut, P., & McMillan, S. L. W. 1996, ApJ, 467, 359
Heggie, D. C., & Mathieu, R. D. 1986, in The Use of Supercomputers in Stellar Dynamics, ed. P. Hut & S. L. W. McMillan (Berlin: Springer), 233
Heggie, D. C., & Stevenson, D. 1988, MNRAS, 230, 223
Hénon, M. 1971a, Ap&SS, 13, 284
———. 1971b, Ap&SS, 14, 151
Hernquist, L., Hut, P., & Makino, J. 1993, ApJ, 402, L85
Hut, P., & Makino, J. 1999, Science, 283, 501
Hut, P., McMillan, S., Goodman, J., Mateo, M., Phinney, E. S., Pryor, C., Richer, H. B., Verbunt, F., & Weinberg, M. 1992b, PASP, 104, 981
Hut, P., McMillan, S., & Romani, R. W. 1992a, ApJ, 389, 527
Joshi, K. J., Nave, C., & Rasio, F. A. 2000, ApJ, submitted (astro-ph/9912155)
King, I. R. 1966, AJ, 71, 64
Kochanek, C. S. 1992, ApJ, 385, 604
Kumar, P., & Goodman, J. 1996, ApJ, 466, 946
Lightman, A. P., & Shapiro, S. L. 1977, ApJ, 211, 244
Lynden-Bell, D., & Eggleton, P. P. 1980, MNRAS, 191, 483L
Makino, J. 1996, ApJ, 471, 796
Makino, J., Taiji, M., Ebisuzaki, T., & Sugimoto, D. 1997, ApJ, 480, 432
McMillan, S. L. W., & Hut, P. 1994, ApJ, 427, 793
McMillan, S. L. W., Hut, P., & Makino, J. 1990, ApJ, 362, 522
———. 1991, ApJ, 372, 111
Murphy, B., & Cohn, H. 1988, MNRAS, 232, 835
Piotto, G., Zocalli, M., King, I. R., Djorgovski, S. G., Sosin, C., Dorman, B., Rich, R. M., & Meylan, G. 1999, AJ, 117, 264
Portegies Zwart, S. 1995, Ph.D. thesis, Univ. Utrecht
Portegies Zwart, S., Hut, P., Makino, J., & McMillan, S. L. W. 1998, A&A, 337, 363
Portegies Zwart, S., Hut, P., McMillan, S. L. W., & Verbunt, F. 1997, A&A, 328, 143
Portegies Zwart, S., Makino, J., McMillan, S. L. W., & Hut, P. 1999, A&A, 348, 117
Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. 1992, Numerical Recipes in C: The Art of Scientific Computing (2nd ed.; New York: Cambridge Univ. Press)
Quinlan, G. D. 1996, NewA, 1, 255
Robinson, C., Lyne, A. G., Manchester, R. N., Bailes, M., D'Amico, N., & Johnston, S. 1995, MNRAS, 274, 547
Rubenstein, E. P., & Bailyn, C. D. 1997, ApJ, 474, 701
Sigurdsson, S., & Phinney, E. S. 1995, ApJS, 99, 609
Sosin, C., & King, I. R. 1995, AJ, 109, 639
———. 1997, AJ, 113, 1328
Spitzer, L. 1987, Dynamical Evolution of Globular Clusters (Princeton: Princeton University Press)
Spitzer, L., & Hart, M. H. 1971a, ApJ, 164, 399
———. 1971b, ApJ, 166, 483
Spurzem, R., & Giersz, M. 1996, MNRAS, 283, 805
Statler, T. S., Ostriker, J. P., & Cohn, H. 1987, ApJ, 316, 626
Stodółkiewicz, J. S. 1982, Acta Astron., 32, 63
———. 1985, in IAU Symp. 113, Dynamics of Star Clusters, ed. J. Goodman & P. Hut (Dordrecht, Reidel), 361
———. 1986, Acta Astron., 36, 19
Sugimoto, D., & Bettwieser, E. 1983, MNRAS, 204, 19P
Takahashi, K. 1993, PASJ, 45, 233
———. 1995, PASJ, 47, 561
———. 1996, PASJ, 48, 691
———. 1997, PASJ, 49, 547
Takahashi, K., & Portegies Zwart, S. F. 1998, ApJ, 503, L49
———. 2000, ApJ, 545, 759