THE ASTROPHYSICAL JOURNAL SUPPLEMENT SERIES, 70:389–417, 1989 June

ERROR ANALYSIS OF A TREE CODE

JOSHUA E. BARNES AND PIET HUT
Institute for Advanced Study
Received 1988 July 18; accepted 1988 October 10

ABSTRACT

Recently several groups have described novel algorithms for solving the gravitational N-body problem with computing time of order $N \log N$ or N instead of N^2. These algorithms are known as "tree codes" because a tree-shaped data structure is used to organize the particles into a hierarchy of even larger groups. Like most numerical schemes, tree codes are *approximate* methods, delivering increasing accuracy as more computer time is invested. Since the approximations involved are novel and therefore unfamiliar, tree codes have only just begun to catch on in stellar dynamics.

Here we present a detailed analytic and numerical investigation of the error properties and performance of one such tree code, which is based on a hierarchical partition of space into cubical cells. Our analysis explores three different levels. We first consider the effect of replacing a distant cell containing many particles with a single point mass and find that the scaling of errors with opening angle and particle number can be well understood theoretically. We next discuss the net effect of many such errors on the total force on a single particle within a large-N system; making analytic estimates is difficult because errors from different cells can add coherently, but numerical tests show that the total error is well-behaved and can be made quite small at a reasonable computational cost. Finally, we consider the cumulative effect of force-calculation errors on the trajectories of individual particles and the dynamical evolution of N-body systems, focusing on a set of tests designed to assess how reliable tree codes are for self-consistent collisionless problems. We conclude that as a general class tree codes appear to be the method of choice when modeling collisionless three-dimensional N-body systems with arbitrary geometry. C and Fortran 77 implementations of our algorithm are available on request.

Subject headings: numerical methods

I. INTRODUCTION

Broadly speaking, an N-body algorithm has two components: a method for evaluating the force on particles and a method for integrating the orbits of particles. Orbit integrators have approached a local optimum with Aarseth's (1985) implementation of Ahmad and Cohen's (1973) neighbor scheme within his individual-time-step algorithm. In addition, Aarseth has developed codes that include the regularization of many-body encounters, which is vital for simulations of collisional systems such as globular clusters. In contrast, force calculators have evolved in many directions, with simple direct-sum $O(N^2)$ algorithms augmented by optical analog methods (Holmberg 1941), particle-mesh algorithms (Miller 1978; Hockney and Eastwood 1981), particle-particle particle-mesh methods (Hockney and Eastwood 1981; Efstathiou *et al.* 1985), global multipole expansion techniques (van Albada 1982; Villumsen 1982), and recently various hierarchical algorithms (Appel 1981, 1985; Jernigan 1985; Porter 1985; Barnes and Hut 1986; Greengard and Rokhlin 1987; Press 1986). Most of these methods offer fast *approximation* of the gravitational field for large-N systems and so are well suited to collisionless problems, which typically place more priority on a large value of N than on very precise orbit integration. However, most fast force calculators also set rather frustrating constraints on the mass distributions allowed, discouraging many interesting projects. For example, three-dimensional particle-mesh codes can model only a modest range of length scales, while improved schemes that include local particle-particle interactions slow down if large density contrasts develop. Global expansion techniques get around these problems, but can only handle rather smooth spheroidal distributions with one or two density centers. At present, only hierarchical methods seem promising for *general* three-dimensional collisionless calculations.

Hierarchical methods are based on the observation that, when calculating the gravitational force acting on a given particle, it makes sense to ignore the detailed internal structure of distant groups of many particles. By replacing many similar particle-particle interactions with a single particle-group interaction, which may include low-order corrections for the group's internal structure, hierarchical algorithms achieve very significant computational savings. At first sight it appears more expensive to identify suitable groupings of distant particles than to simply calculate all interactions directly. The trick is to organize the particles into a tree-shaped data structure, which partitions the mass distribution into a hierarchy of suitable groups, and is repeatedly updated as the system evolves. Recursive "descent" of the tree structure can yield a good approximation to the total force on a single particle in only $O(\log N)$ steps.[1]

[1] Computer scientists draw their trees upside down, so descent means moving from the root toward the leaves. This should seem perfectly natural to astronomers, who assign large magnitudes to stars with small luminosities.

The algorithm developed by Barnes and Hut (1986), which is described in more detail in Appendix A, follows the general approach described above, but with one important twist. Tree codes developed by Appel (1981, 1985), Jernigan (1985) and Porter (1985), and Press (1986) used tree structures that try to conform to the (presumably) hierarchical mass distribution of the system. Such trees may be more trouble than they are worth, because the faithfulness of the tree becomes a critical issue in the analysis of the code. In contrast, we based our tree on the idea of recursively refining regions of *space*. The interior nodes of the tree correspond to cubical cells, while particles themselves correspond to the leaves at the ends of the branches. Starting with the "root" cell, which encloses the entire system, we repeatedly subdivide any cell containing more than one particle into eight smaller subcells, continuing until all particles are isolated in individual cells. Tree construction is concluded by propagating mass and center-of-mass information back from the leaves toward the root cell. The entire construction process requires $O(N \log N)$ steps and typically takes only ~2% of the CPU time of an equal-time-step algorithm running on a workstation or mainframe; on a vectorizing supercomputer a comparable level of performance can be obtained with careful coding (Makino 1989; Hernquist 1989).

Given this tree structure, the recursive descent procedure computes the force on a particle by constructing a description of the surrounding mass distribution which is fine grained locally but increasingly coarse grained at longer ranges. We start at the root cell of the tree and descend toward the leaves according to the following rules. If we encounter a particle or a cell which subtends less than a critical linear angle θ as seen from the particle on which we are currently evaluating the force we compute the particle-particle or particle-cell interaction and add the result to the running sum for the current particle. Otherwise, we have presumably encountered a cell which is too close to the current particle to be treated as a unit, so instead of calculating a particle-cell interaction directly, we recursively examine the subcells of this cell. Essentially, this procedure guarantees (1) that all cells interacting with the current particle subtend linear angles of less than θ and (2) that each particle in the system is counted once, either directly in a particle-particle interaction or indirectly via a particle-cell interaction involving one and only one enclosing cell. A simple argument given by Barnes and Hut (1986) shows that this tree-search procedure is $O(\log N)$; hence the force on all N particles can be computed in $O(N \log N)$ steps.

a) *Error Analysis*: *Motivation and an Example*

A head-on collision between two identical spherical systems illustrates the kind of systematic errors which can crop up with a tree algorithm. In the experiment presented here, two N-body realizations of standard King (1966) models, with unit mass $M = 1$, binding energy $E = -1/2$, and dimensionless central potential $W_c = 5$, are generated using $N = 2048$ bodies each. The two "galaxies" are launched at each other from a distance of two length units, with the velocity they would acquire falling from rest at infinity. The equations of motion are integrated by a simple time-centered leapfrog with time step $\Delta t = 0.025$, using the tree force-calculation algorithm with opening angle $\theta = 1$ and force softening $\varepsilon = 0.025$.

Figures 1*a*–1*c* present three "snapshots" of the system, showing the initial conditions ($t = 0$), the configuration near maximum overlap ($t = 1$), and the configuration shortly before the final merger ($t = 3$). The overall behavior of this simulation is very similar to that found in previous studies of head-on collisions using other techniques (e.g., van Albada and van Gorkom 1977). As the two galaxies overlap, the total mass within a given radius r doubles, so each particle is in effect perturbed by a strong radial impulse, lasting a fraction of the local dynamical time. When the galaxies separate, they "rebound," launching a significant part of their mass on loosely bound or even unbound orbits; the energy required to do this comes from their relative motion, which therefore decays.

Figure 2 shows the fractional variation in the computed binding energy of the system as a function of time, $[E(t) - E_0]/E_0$, where $E_0 \simeq -1.0$ is the value of the binding energy at $t = 0$. Dots indicate measurements made using the tree algorithm to compute potentials, while circles indicate occasional (and expensive) checks made using a direct-sum calculation. Two systematic effects are apparent in this figure:

1. The tree algorithm underestimates the depth of the potential well by 0.3%–0.8%, as shown by the net offset of the circles.

2. Conservation of energy is violated at the ~2% level. At maximum overlap the binding energy of the system, E, is slightly too negative, and when the system reexpands the opposite effect is seen.

The second effect could conceivably be due to either the force or integration algorithms, so two additional calculations, one with $\Delta t = 0.0125$ and one with $\theta = 0.7$, were run. Reducing the time step by a factor of 2 had almost no effect on energy conservation. Refining the force calculation so as to double the average number of interactions calculated per particle, however, reduced the violation of energy conservation to slightly less than 1%, proving that the force calculation is the culprit.

b) *Long-Range Goals and Short-Range Plans*

How can we establish the correctness of an N-body calculation? Traditionally, most practitioners have used conservation of total binding energy and, to a lesser extent, total linear and angular momentum as the primary measures of accuracy. This practice is perhaps based on the general expectation that only a very small subspace of possible numerical errors could seriously compromise the results of an N-body calculation without affecting E and $\boldsymbol{J}$. But, in fact, *any* dynamical system based on a spherically symmetric two-particle interaction potential will conserve total momentum and energy, no matter how much the potential differs from the Keplerian $1/r$ form. Global conservation may be less reliable than one would like. If more stringent tests of N-body models are available, they should be applied as well.

At this point we must distinguish between the mathematical and numerical descriptions of the N-body problem. From a mathematical point of view, the N-body equations of motion define a mapping from the initial to the final states of the system which is *continuous* everywhere, except for a singular

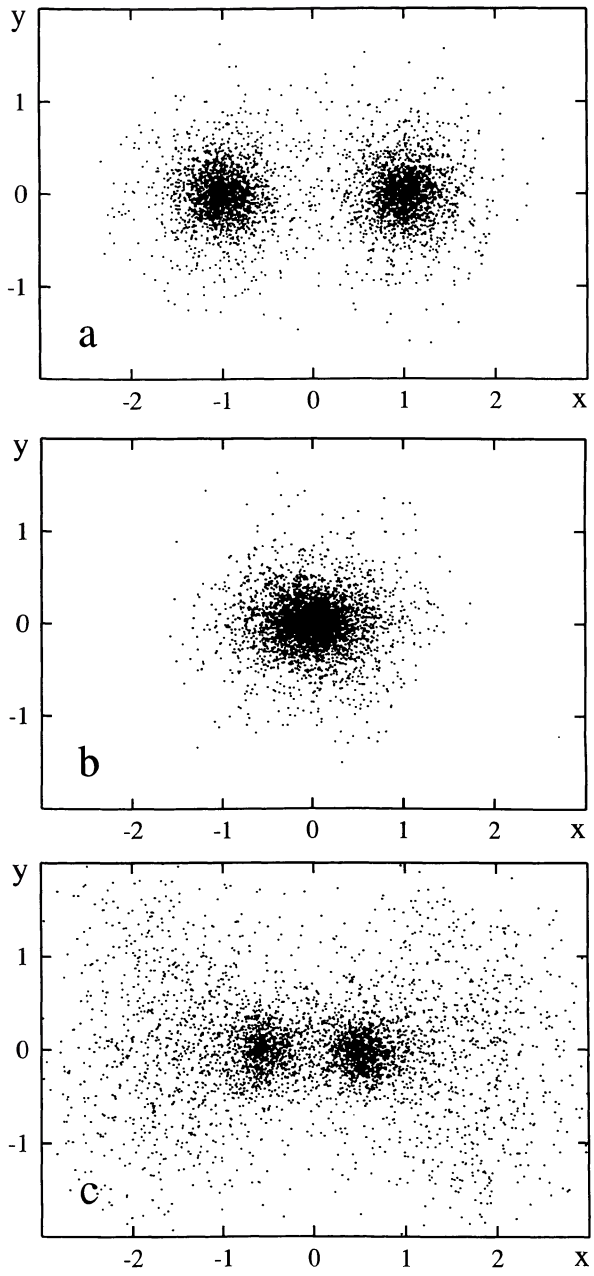


FIG. 1.—Head-on collision of two identical King models with central potentials $W_c = 5$, run with the BH tree algorithm. (*a*) Initial conditions, $t = 0$; (*b*) first encounter, $t = 1$; (*c*) between first and second encounters, $t = 3$.

set of measure zero. However, from a numerical point of view we may not be able to demonstrate the continuity of this mapping. Some systems depend sensitively on bits which were too insignificant to be specified in initial conditions of reasonable word length (Miller 1964; Dejonghe and Hut 1986; Heggie, Goodman, and Hut 1989). These missing bits are effectively supplied by the N-body integrator, which acts as a source of microscopic noise; of course, different machines and integration algorithms inject entirely different noise patterns. Unless the N-body code introduces a bias into the evolution of the system, for example by avoiding multibody encounters, these low-level noise sources should not seriously compromise the results. But as Lecar (1968) concluded, the rapid divergence of nearby trajectories means that we cannot establish the correctness of typical collisional simulations by comparing the motions of corresponding particles in different calculations starting from the same initial conditions. Collisional simulations can only be tested by observing the convergence of *statistical* measures derived from coordinates of many particles and by noting good conservation of E and $\boldsymbol{J}$ together with an accurate $1/r^2$ force field.

In other cases, when the Lyapunov coefficients of the system are not too large and the calculation runs for only a few dynamical times, we may be able to demonstrate the continuity of the N-body mapping numerically by showing that a small perturbation in the initial conditions leads to a small perturbation in the final state. The solution supplied by an N-body code can then be viewed as an approximation to the canonical trajectory of the system. There are many ways to assess the accuracy of this approximation. Starting from a given set of initial conditions, we can run a sequence of calculations, successively refining nonphysical parameters such as the time step Δt, force-softening parameter ε, or critical opening angle θ. If the trajectories of corresponding particles in different runs along this sequence converge to a definite limit, before diverging due to round-off errors,[2] we can conclude that the N-body integration provides a reliable mapping from initial to final state. On the other hand, we can also run a sequence of calculations with different total N, but with some common subset of particles starting from the same place in phase space in each case. If these common particles follow similar trajectories, regardless of N, we can conclude that relaxation is not a problem. Together, convergence and relaxation tests can provide strong support for the correctness of a collisionless calculation. Moreover, by identifying the dominant source of error in a calculation, they can guide us in selecting the most efficient choice of parameters Δt, ε, θ, and N for a given physical problem, minimizing the computing time while maintaining physical accuracy.

Before embarking on such a program of "throughput" tests, however, we would like to first examine the detailed performance of our tree force calculator in terms of computing time required and statistical distribution of errors made in force approximation, as a function of opening angle θ, softening ε, particle number N, and distribution of particles. A completely analytic treatment of this problem is too hard; for starters, we would need a statistical description of the mass distribution on both microscopic and macroscopic scales and a theory of how to sum the partially correlated errors due to many particle-cell interactions. On the other hand, a numerical treatment is completely straightforward: create an N-body system, feed it to the tree-code force algorithm, and compare the results with an N^2 calculation. This kind of empirical approach has already been used by Hernquist (1987) to study the Barnes-Hut (BH) algorithm; the results which we will present in § III*b* complement and extend his work.

In this paper we discuss force-calculation errors on three levels. First, in § II we consider the local static error made when some complicated but distant set of particles is repre-

[2]In principle, one could use a computer with adjustable word length, such as the Connection Machine 1 described by Hillis (1985), to directly check round-off errors.

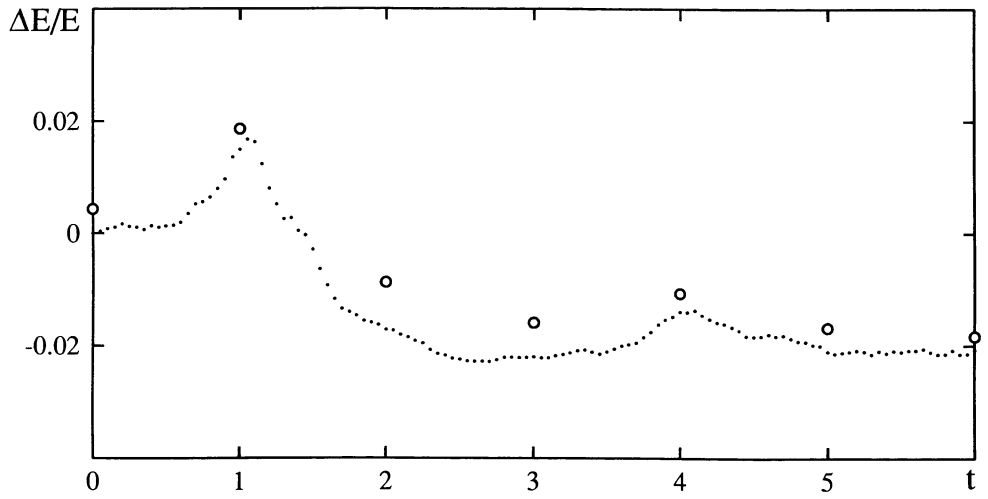


FIG. 2.—Total binding energy of the system shown in Fig. 1, plotted as a function of time. *Dots*: potential energy computed using tree algorithm; *circles*: potential energy computed using an $O(N^2)$ algorithm. The peaks near $t=1$ and $t=4$ correspond to the first and second collisions between the two galaxies.

sented by a single point mass. This is a problem which *can* be addressed analytically, as is illustrated in great detail in Appendices B and C. We then compare such calculations with numerical experiments, confirming the analytic theory but also illustrating the complicated interplay of effects due to finite particle number. Next, in § III we consider the error made by the BH tree code in evaluating the *total* force on a single particle in a larger mass distribution. Here it is possible to predict the general behavior of the force errors, but detailed results require a numerical treatment; we give sample results, but defer a detailed description of methods and results to Appendix D. Finally, in § IV we study the dynamical errors made when running an N-body code, focusing on the problem of realistically simulating the collisionless flow of particles in phase space. We contrast the effect of force-calculation errors with other errors due to integration, softening, and the discrete sampling of a smooth mass distribution. We conclude in § V that presently implemented tree codes can indeed provide reliable numerical simulations of collisionless systems, since there is now a well-understood procedure to obtain arbitrary accuracy in an efficient and controlled way.

II. ERROR FOR A SINGLE BODY-CELL INTERACTION

The potential at the position of a given particle, due to the gravitational field of all other particles, can be determined approximately by our tree algorithm. This approximation process introduces an error, which consists of the sum of neglected higher order multipole moment contributions of all cells containing more than one particle and contributing to the force calculation for the given particle. In the simplest case, the force contribution from a cell is determined by putting all its mass in the center of mass of the original mass distribution. This procedure guarantees that the monopole moment of the cell is represented correctly and that the dipole moment vanishes. The resulting error per cell is given by the sum of quadrupole and higher multipole moments of the particle distribution in that cell.

a) Analytic Theory for $N \to \infty$

A major advantage of our tree method of force approximations is that we have a direct and precise handle on the range of errors introduced by our approximations. In the simplest monopole-only version of our tree method, we face a straightforward, albeit somewhat tedious, exercise in classical potential theory: given the mass distribution in a cubical cell, compute the successive multipole contributions to the force at a given point outside the cell.

Let us start with the simplest case, in which the cell contains a homogeneous mass distribution. In this case the quadrupole and octopole moments vanish identically, as can be argued from the lack of flattening and "pear-shapedness" of a cube and as is demonstrated by direct inspection in Appendix B. The leading term is formed by the next multipole moment, the hexadecapole moment. As derived in Appendix B, we find for the gravitational potential of a homogeneous cube, up to hexadecapole moment

$$\Phi(\boldsymbol{r}) = \frac{GM}{r} + \frac{7}{960}\frac{GMa^4}{r^9}\left[3r^4 - 5\left(x^4 + y^4 + z^4\right)\right], \quad (1)$$

where the potential $\Phi(\boldsymbol{r})$ is defined with the opposite sign of the usual definition, G is the gravitational constant, $\boldsymbol{r} = (x, y, z)$ is the position vector of the observer, as seen from the center of the cell, $r = |\boldsymbol{r}|$, and a is the length of the side of the cube, centered on the origin of a Cartesian coordinate system. M is the total amount of mass in the cell.

The gravitational force measured by an observer at $\boldsymbol{r}$ follows directly from the potential (eq. [1]). For example, the x-component of the force, up to hexadecapole expansion, is given by

$$F_x(\boldsymbol{r}) = \frac{M}{r^3}x - \frac{7}{96}\frac{ma^4}{r^{11}}x\left[5x^2r^2 - 6x^4 - 3\left(y^4 + z^4\right)\right]. \quad (2)$$

More accurate approximations can be obtained in a similar way. In Appendix B we present expressions for the next term in the multipole expansion, the 64-pole (hexacontatetrapole?)

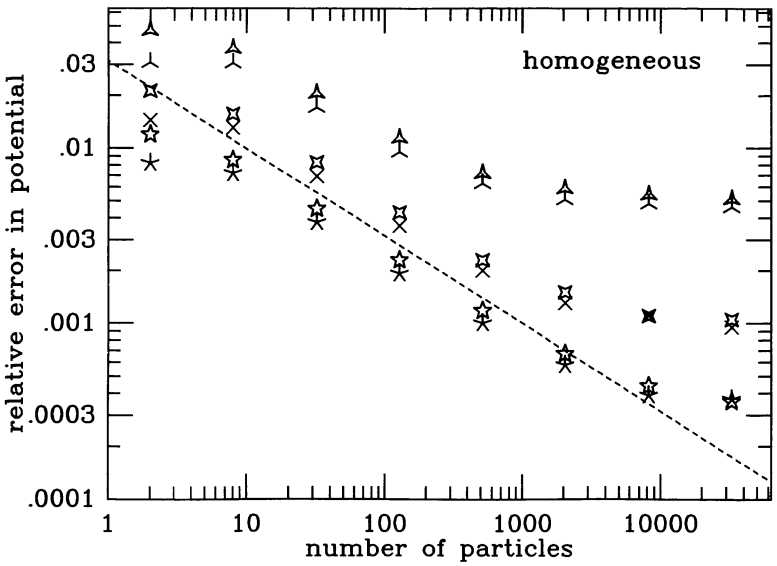


FIG. 3.—Fractional errors in the gravitational potential, measured outside a cell which contains N particles ($N = 2$, 8, 32, 128, 512, 2048, 8192, and 32,768), drawn randomly from a homogeneous distribution. For each N value, three values are chosen for the distance of the observer, to reproduce opening angles $\theta = 1$ (*upper two symbols*; *three-pointed stars*), $\theta = 2/3$ (*middle two symbols*; *four-pointed stars*), and $\theta = 1/2$ (*lower two symbols*; *five-pointed stars*). For each $\{N, \theta\}$ combination, an open star indicates the average and a skeletal star indicates the median of a large number of different experiments with different random particle distributions and different random orientations of the cell with respect to the observer. The diagonal line has been added to indicate a slope proportional to $\sqrt{N}$, at an arbitrary offset.

contribution, both for the potential (eq. [B28]) and the force (eq. [B29]).

The next exercise, after having gained a detailed understanding of the expansion of the potential of a homogeneous cube, is the introduction of a linear density gradient. The computations, while still elementary, now become exceedingly tedious. The leading error term is derived in Appendix C. In terms of the potential, we find

$$\Phi(\boldsymbol{r}) = \frac{GM}{r}\left[1 + \frac{1}{288}\left(\frac{a}{r}\right)^2\left(\frac{\rho_1'}{\rho_0'}\right)^2\left(1 - 3\frac{(\boldsymbol{r}\cdot\hat{\boldsymbol{n}})^2}{r^2}\right)\right]. \quad (3)$$

Here $\hat{\boldsymbol{n}}$ is a unit vector pointing along the density gradient. Specifically, the density is defined as

$$\rho(\boldsymbol{r}) = \rho(x, y, z) = \begin{cases} \rho_0 + (1/a)(\boldsymbol{r}\cdot\hat{\boldsymbol{n}})\rho_1, & |x| < \frac{1}{2}a,\ |y| < \frac{1}{2}a, \\ & \text{and } |z| < \frac{1}{2}a, \\ 0, & \text{elsewhere}, \end{cases} \quad (4)$$

where ρ_0 and ρ_1 are constants, setting the scale for the underlying homogeneous density and the linear gradient, respectively.

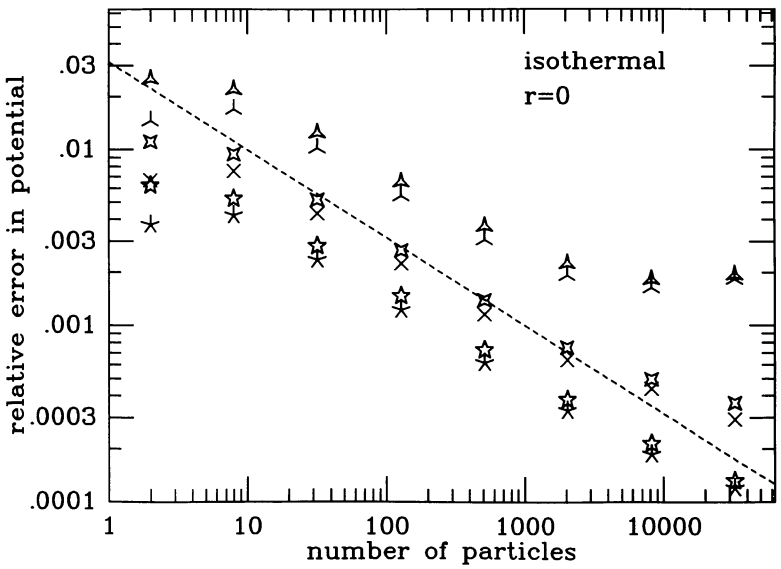


FIG. 4.—Like Fig. 3, but for a distribution of particles which is centrally concentrated according to a singular isothermal distribution $\rho \propto r^{-2}$. The density center $r = 0$ coincides with the center of the cell.

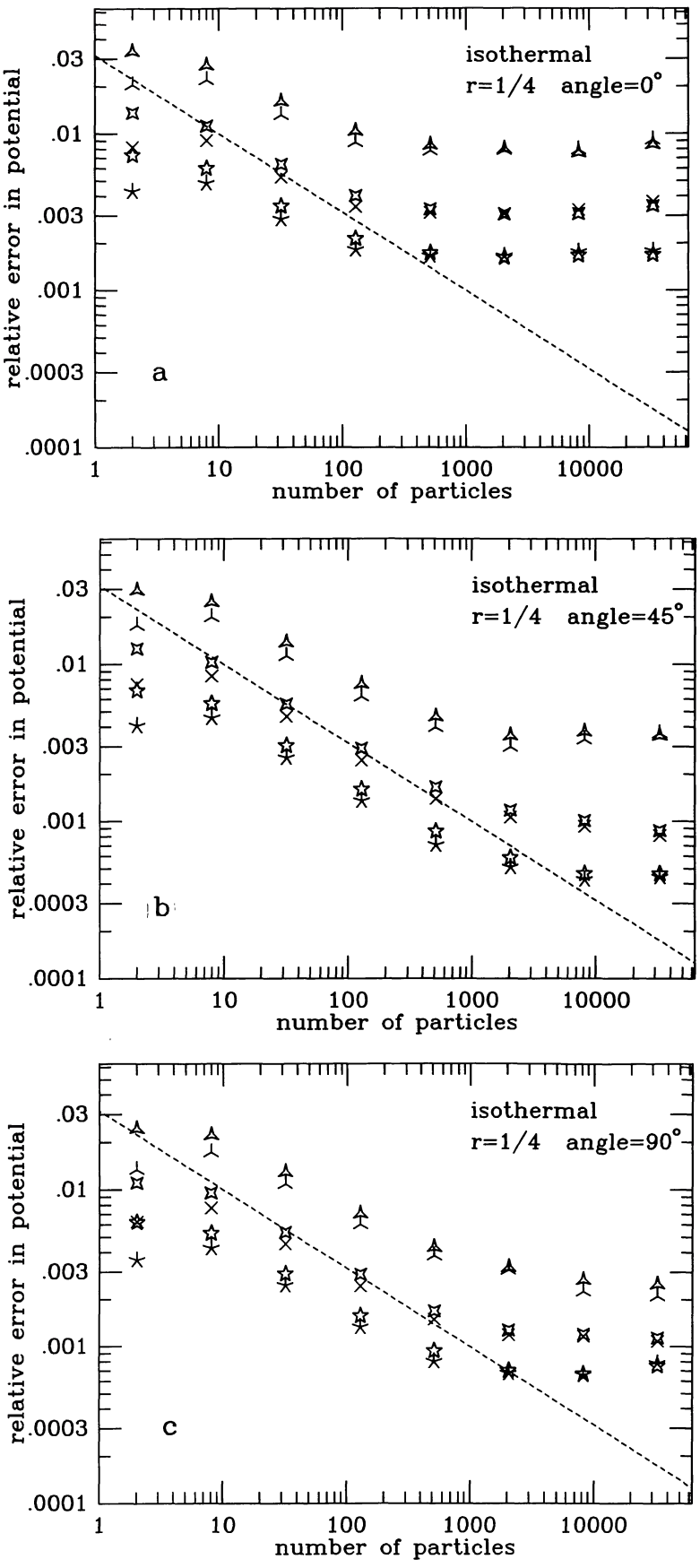
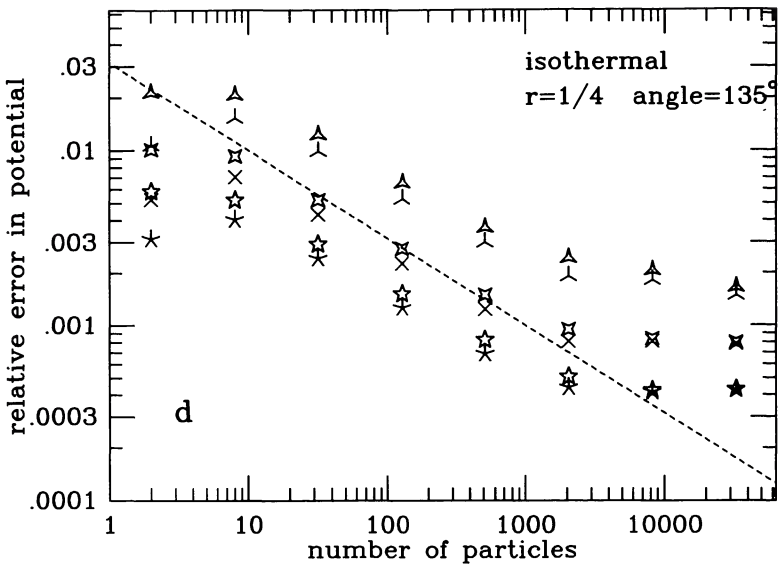


FIG. 5.—Like Fig. 4, but with a density center which has an offset of $r = 1/4$ away from the center of the cell. Five sets of measurements are shown, each for a different value of the angle β between the observer and the center of the cluster as seen from the center of the box.



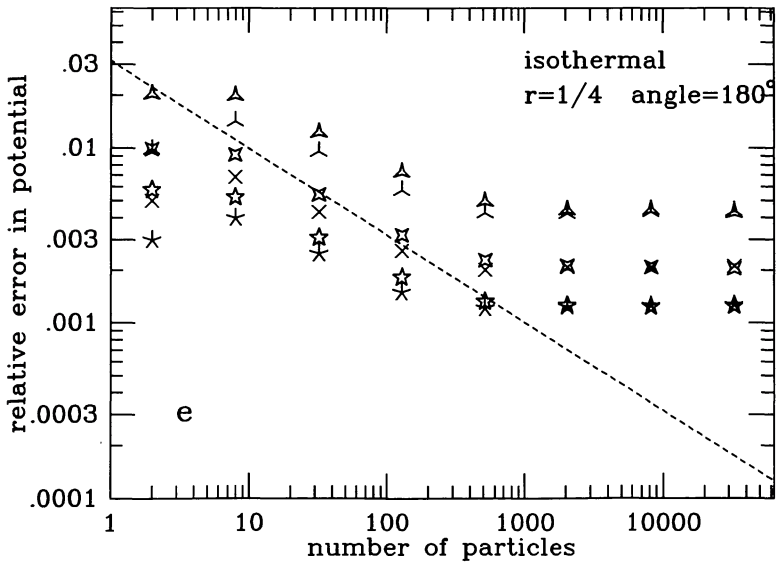


FIG. 5—*Continued*

In Appendix C, we present the contributions of the next two terms in the expansion, the octopole and the hexadecapole moments. The reason for extending the expansion that far is to make contact between the leading error in the homogeneous case and the extra errors introduced by an additional linear density gradient. The results for the potential are given in equations (C19) and (C22). Unfortunately, the expressions turn out to be unwieldy, and they are therefore mainly of interest for applications through numerical evaluations. This brings us to our next exercise, namely a direct numerical determination, in the form of Monte Carlo–type experiments, of the error contributions from a single box, up to all orders in the expansion.

b) *Numerical Experiments for Finite N*

The first set of numerical tests was performed for the case of a homogeneous distribution of particles. The recipe for the tests is as follows. Consider a three-dimensional box with sides of unit length. Sprinkle N particles in the box at random positions. Measure the gravitational potential and the gravitational force at a distance r_o from the center of the box (r_o stands for the position of the observer). The only extra parameter that needs to be specified is the angle between the observer and the orientation of one of the sides of the box. To simplify the presentation of the results, we have performed many different experiments for different random values of

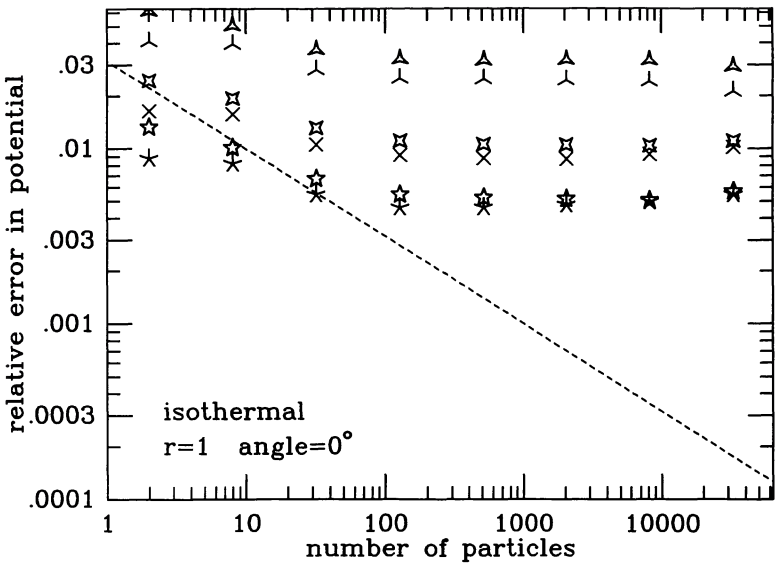


FIG. 6*a*

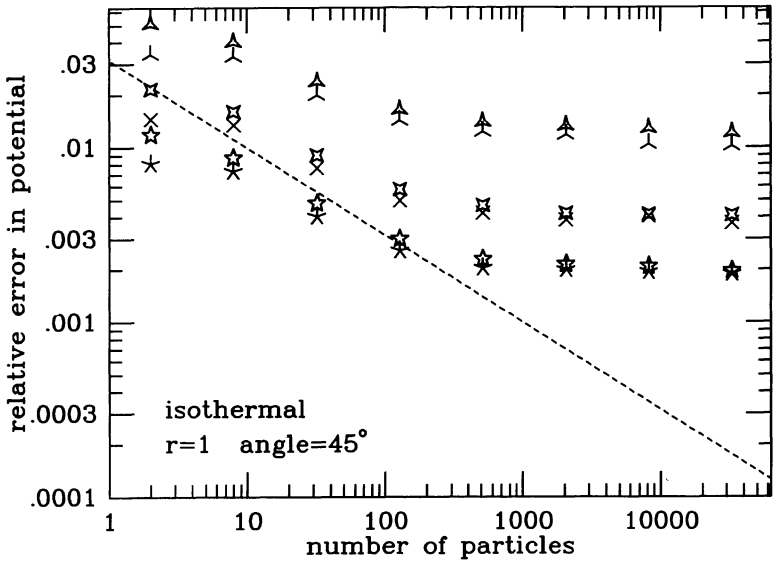


FIG. 6*b*

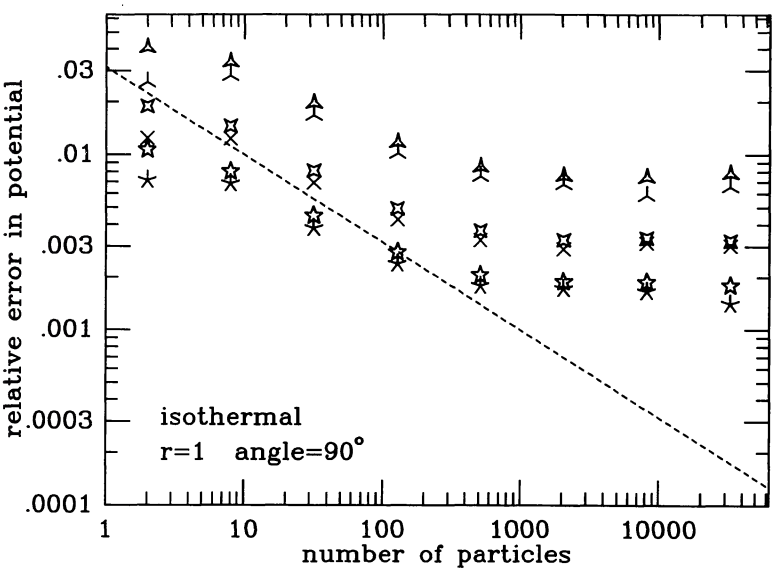


FIG. 6*c*

FIG. 6.—Like Fig. 5, but with a density center offset $r = 1$

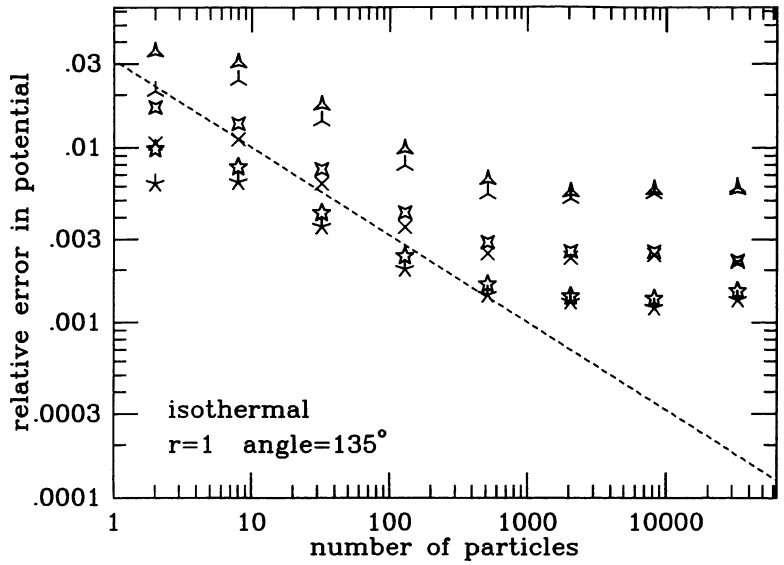


FIG. 6*d*

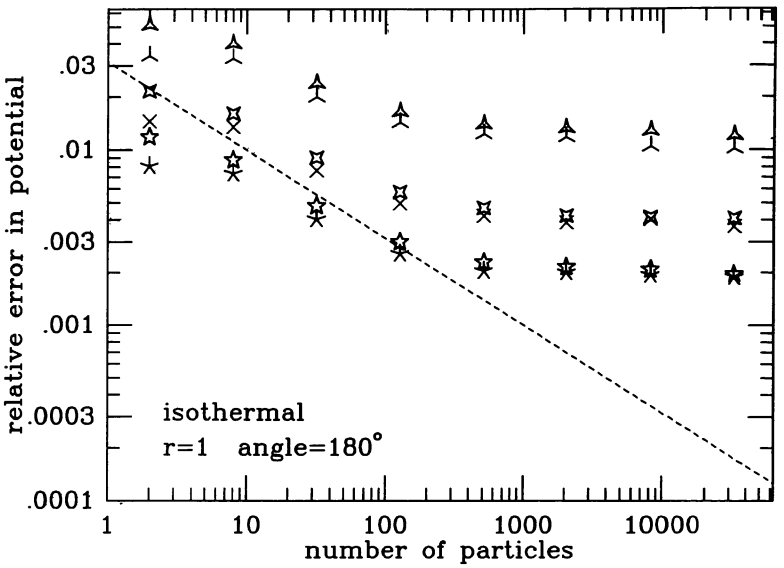


FIG. 6*e*

this angle. In Figure 3 we plot the main results of each such series of experiments by two different symbols, open stars for the average values and skeletal stars for the median values of the distribution of measurements for a single setting of the N and r_o dials. As can be seen in Figure 3, the averages always tend to be somewhat larger than the medians because the contribution of occasional larger errors to the average is proportional to their magnitude, while their contribution to the median is independent of their magnitude.

Results in Figure 3 are given for $N = 2$, 8, 32, 128, 512, 2048, 8192, and 32,768. The subdivision parameter θ, introduced in § I, determines the size of the largest cell which is not subdivided further, in units of the cell distance (Barnes and Hut 1986). In these experiments $\theta = 1/r_o$, since the box size is unity. For each N value, the three pairs of symbols in Figure 4 indicate the results of measurements for different θ values. The uppermost symbols, which indicate the largest errors, give the results for $r_o = 1$ or equivalently $\theta = 1$. The middle symbols apply to $r_o = 1.5$ or $\theta = 2/3$. The lower symbols indicate the results for $r_o = 2$ or equivalently $\theta = 0.5$. To further highlight the difference in θ values, we use three-pointed stars to indicate $\theta = 1$, four-pointed stars to indicate $\theta = 2/3$, and five-pointed stars to indicate $\theta = 1/2$.

The results for the homogeneous case can be understood in detail from the analytical derivations given in Appendix B. In the range from $N = 8$ to $N = 128$ the errors scale proportional to $N^{1/2}$, with a constant of proportionality θ^2. The scaling with N follows from the fact that the leading error contribution stems from the quadrupole contribution. In the limit $N \to \infty$ the quadrupole moment of a homogeneous solid cube

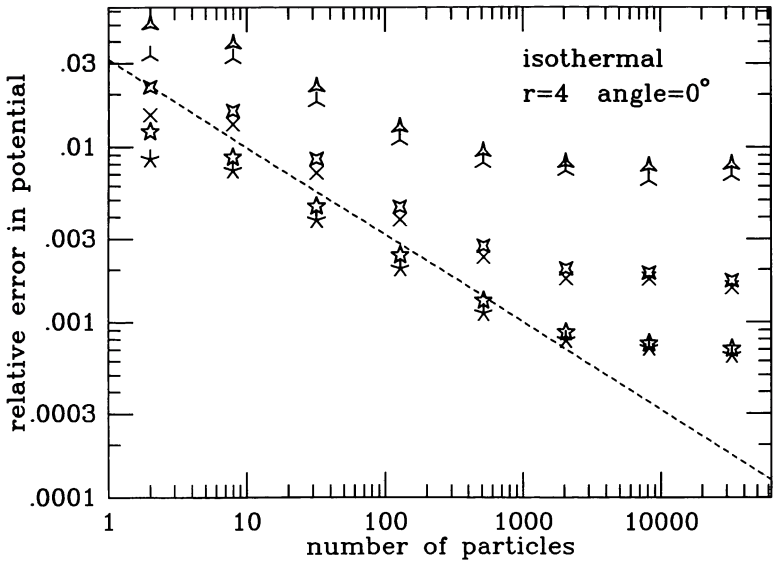


FIG. 7*a*

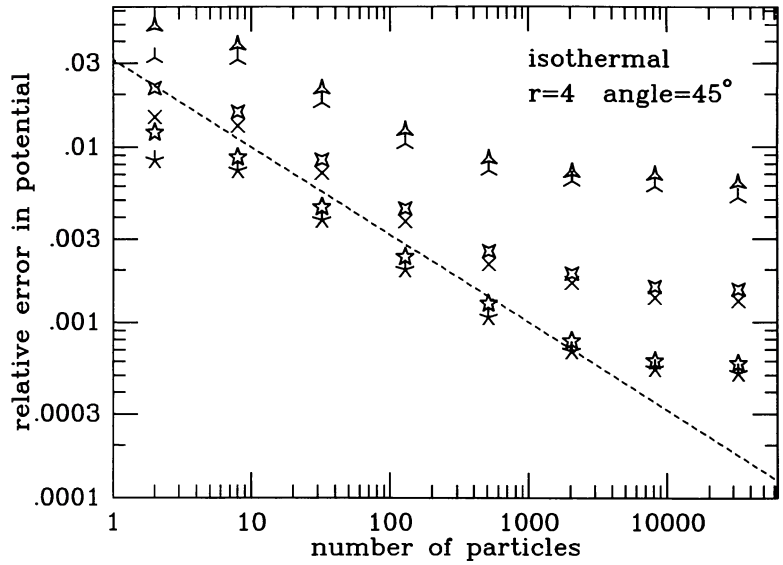


FIG. 7*b*

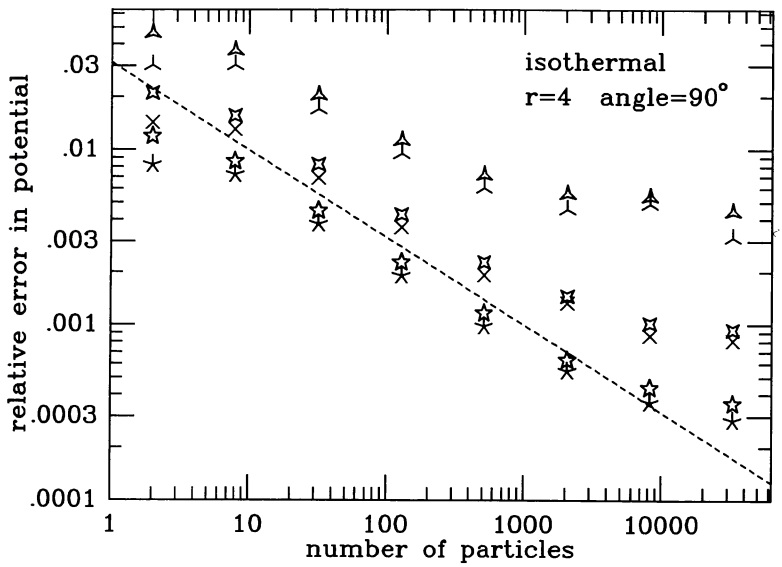


FIG. 7*c*

FIG. 7.—Like Fig. 5, but with a density center offset $r = 4$

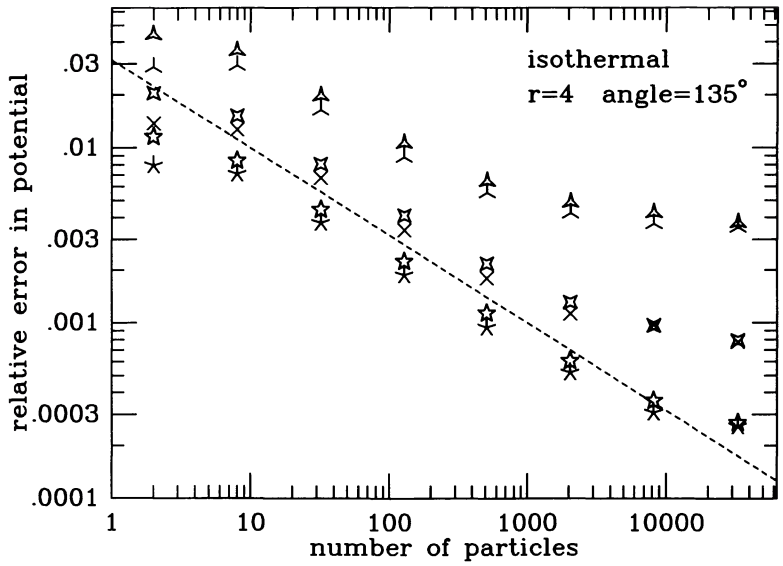


FIG. 7*d*

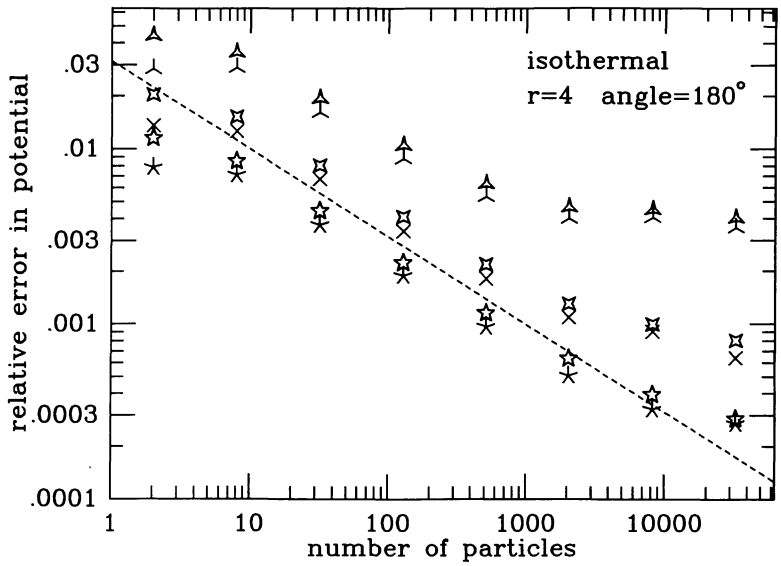


FIG. 7*e*

vanishes. For finite N, therefore, the quadrupole moment is entirely due to random fluctuations in particle positions within the box, which average out to a net quadrupole moment proportional to $\sqrt{N}$. To highlight this dependency, we have added a line with slope proportional to $\sqrt{N}$, at an offset which is arbitrary, but which has been kept constant throughout the whole set of figures in this section. The scaling proportional to θ^2 follows from the fact that the quadrupole moment, corresponding to an $l = 2$ spherical harmonic, has to be multiplied by a factor θ^2 in order to determine the contribution to the potential at a fixed distance. For larger N values, $N \gtrsim 10{,}000$, the dominant contribution to the error in the potential arises from the hexadecapole moment, which is the lowest nonvanishing multipole moment of a solid cube. Indeed, this contribution which corresponds to a spherical harmonic with $l = 4$, scales proportional to θ^4, as can be clearly seen for $N = 32{,}768$, where the N dependence has leveled off and nearly reached the asymptotic $N \to \infty$ limit.

As a more realistic particle distribution we now switch from a homogeneous system to a self-gravitating, centrally concentrated system. For this purpose we introduce an extra parameter, α, which characterizes the radial distribution of particles, chosen as a power-law density distribution proportional to $r^{-\alpha}$. Figure 4 shows the result of measuring the potential from a box centered on the density center of the particle distribution. Two differences are immediately evident upon comparing Figures 3 and 4: the errors in Figure 4 are systematically lower, and the hexadecapole saturation sets in for higher particle numbers. The first effect follows from the central concentration of particles inside the box, resulting in

smaller multipole moments of the density distribution. The second effect is related to the lower density near the edges of the box, since the hexadecapole moment is a measure of the "corneredness" of a distribution of particles. Apart from these two major differences, the overall trend remains the same, including the scaling proportional to θ^2 and $\sqrt{N}$ in the fluctuation regime toward the left and proportional to θ^4 and N^0 in the saturation regime.

The next step up in complexity and realism lifts the degeneracy of letting the center of the box coincide with the density center of the particle distribution. Allowing the two to differ introduces two more parameters. First, we have the distance r from the center of the box to the center of a star cluster. In addition, we have the angle β between the observer and the center of the cluster as seen from the center of the box. For symmetry, $0° \leq \beta \leq 180°$. To recapitulate, we now have five dials which we can twist to chose new experiments to perform, one for each of the five parameters, r, r_o, β, N, and α. Figure 5 shows the results for the choice $r = 0.25, \alpha = 2$. Figures 5*a*–5*e* summarize the results for the choices $\beta = 0°$, 45°, 90°, 135°, and 180°, respectively. The effects of changes in N and r_o (or, equivalently, θ) are portrayed within individual frames, as before. Figures 6 and 7 are similar, except for the fact that they show the results for the choices $r = 1$ and $r = 4$, respectively.

Figures 5–7 are less straightforward to interpret. Their behavior cannot be simply explained as being due to quadrupole and hexadecapole contributions. A complex interplay of these two, and octopole contributions as well, occurs which is hard to estimate analytically. In principle, the results of Appendix C enable one to disentangle these effects through analytical approximations, since the expansions in that appendix are carried out to hexadecapole order, at which point the competition between the homogeneous contribution to the hexadecapole moment and the inhomogeneous contributions to the quadrupole and octopole moments has been unraveled. However, even a cursory inspection of equation (C22) gives us sufficient motivation to leave the clarification between the correspondence of equation (C22) and Figures 5–7 as "an exercise for the reader."

III. TOTAL ERROR FOR A SINGLE FORCE CALCULATION

Having studied the error incurred when an arbitrary mass distribution in a distant, cubical cell is replaced by a single point with the same mass and center-of-mass position, we next ask how these errors add up when we compute the total force on a single particle. This would be straightforward if we could assume that the errors due to different cells are statistically independent, but this is not justified. We can only sketch a qualitative theory for the actual error pattern. However, these qualitative arguments can be complemented and illustrated by numerical experiments, which permit a quantitative study of the error properties of a specific algorithm under specific conditions.

a) Qualitative Theory of Total Errors

To comprehend the difficulty of building a theory from first principles for the total force-calculation error, imagine replacing the mass points in an N-body system with equivalent luminosities (see Holmberg 1941). Each point is bathed in a field of radiation, falling off as $1/r^2$, from all others in the system; when a tree-approximation is used to estimate this field, distant groups of lights are replaced by single points of the same total luminosity. Nearby, relatively faint lights surround the observer in an arbitrary pattern, while at greater distances, larger sources are set in nested cubical arrays, ever more perfectly sampled, and from afar a few tremendous "stars" represent huge distant masses. As we move from place to place within the system, these lights hang fixed in space, then suddenly divide into equivalent clusters or merge together into larger units, with nearby ones changing most rapidly, and the most distant ones only as we cross the system. If instead we stand still and watch as particles move around the system, we see nearby particles and small cells jittering and flashing around; more distant sources maintain their identity, but vary in luminosity while shifting around in their cells.

To be more concrete, consider the total error, $\Delta\Phi(x)$, for a homogeneous spherical system, evaluated at points along the x-axis, as shown in Figure 8. This function is piecewise continuous with a finite number of jumps, each one representing a change in the interaction list assigned to a test particle at that point. The extent of a continuous segment is never much longer than the smallest cell on its interaction list, roughly of the order of the interparticle spacing. Within a continuous segment $x_a \leq x < x_b$, the total error is

$$\Delta\Phi(x) = \sum_{j=1}^{n_{pc}} \Delta\Phi_j(x), \tag{5}$$

where $\Delta\Phi_j(x)$ is the error due to cell j and the sum runs over the cells on the interaction list. The error due to an individual cell is generally dominated by the lowest order moment excluded from the interaction calculation (see § II). If this is the quadrupole moment, then $\Delta\Phi_j(x)$ has a term of $O(x^2)$, while if quadrupole moments are included, the lowest nonconstant term is of $O(x^3)$. Within $x_a \leq x < x_b$, the total error in the acceleration, $\Delta\boldsymbol{a}(x)$, is simply the gradient of the potential error:

$$\Delta\boldsymbol{a}(x) = \nabla\,\Delta\Phi = \sum_j \nabla\,\Delta\Phi_j.$$

This relation does not hold at points where the interaction list changes and $\Delta\Phi$ jumps; at such points, $\Delta\boldsymbol{a}$ also exhibits a finite discontinuity.

If the errors from different cells were uncorrelated, we might hope to find some reasonable *Ansatz* for the distribution of $\Delta\Phi_j$ and use the theory of random walks (see Chandrasekhar 1943) to evaluate equation (5) as an incoherent sum. But from the above picture it is clear that we may expect the error from one cell to correlate with the error from its next-door neighbor if both sample part of the same global density field. The quasi-regular lattice of cells seen at intermediate distances can generate significant correlated errors if the

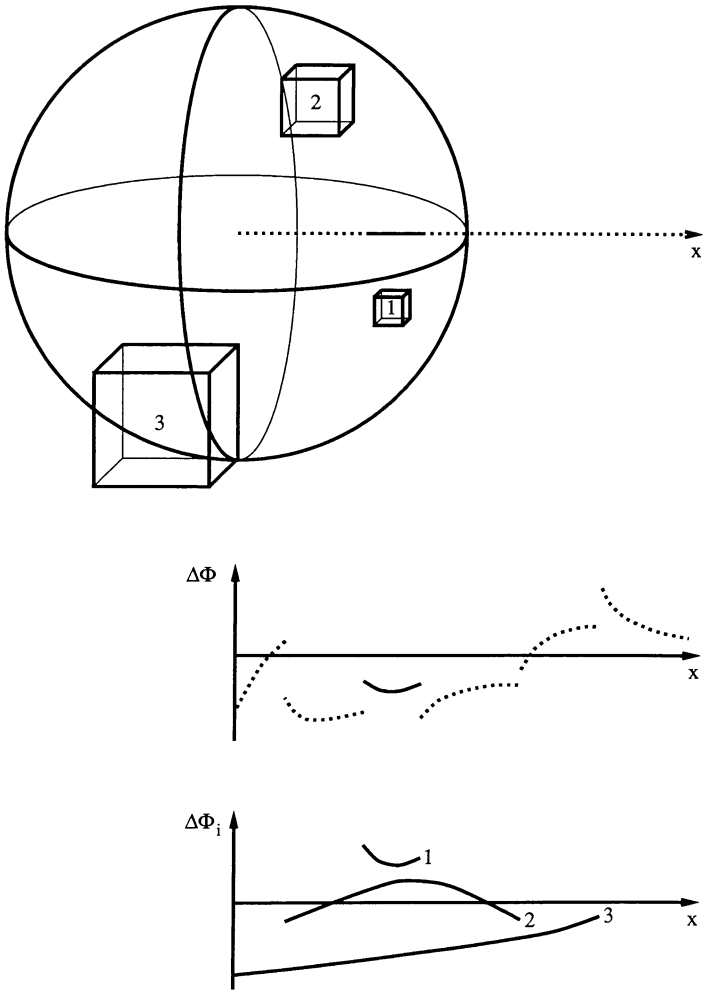


FIG. 8.—Schematic picture of force calculation in a spherical system, sampled at points along the x-axis (*dotted*). The total error is shown as a piecewise continuous function. Three cells on the interaction list of the continuous segment shown as a solid line along the x-axis are indicated, together with their individual contributions to the total error.

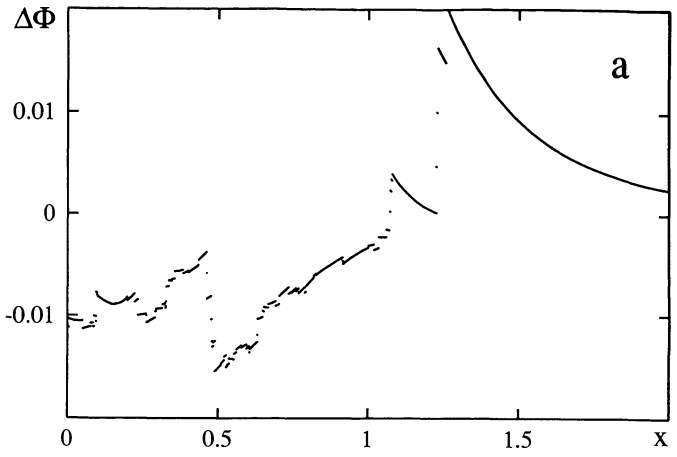
cells cover a region with a coherent large-scale density field, because similar patterns of mass distribution may occur in one cell after another. Furthermore, the error observed at one point is correlated with that found at a nearby point, because some fraction of the total error is due to distant cells common to both observations. In general, we expect some positive correlation in the error field on all length scales up to the scale of the largest cells contributing to the force calculation.

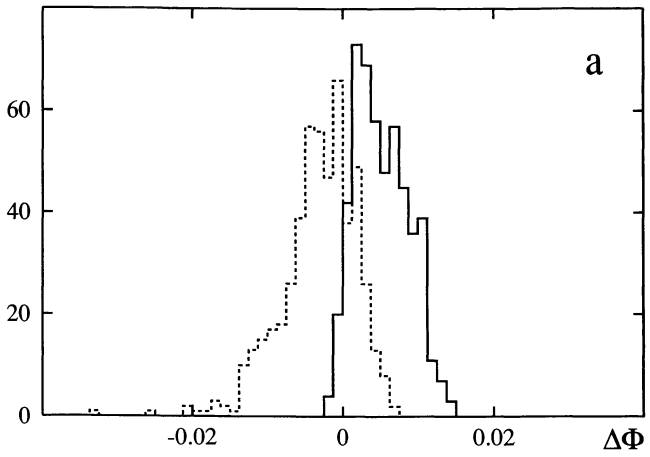
On these last points our analysis diverges sharply from earlier work by Porter (1985), who explicitly assumed that the errors contributed by different nodes are statistically independent. "Lagrangian" tree codes (Appel 1981, 1985; Jernigan 1985; Porter 1985; Press 1986), which organize groups of particles into irregular tree structures, may not generate the same level of correlation between such errors as our nested, regular array of cubical cells does. What is certain is that a rigorous analysis of a Lagrangian tree code is much harder, precisely because the tree is harder to characterize. We suspect that both Eulerian and Lagrangian tree codes produce statistically correlated error patterns.

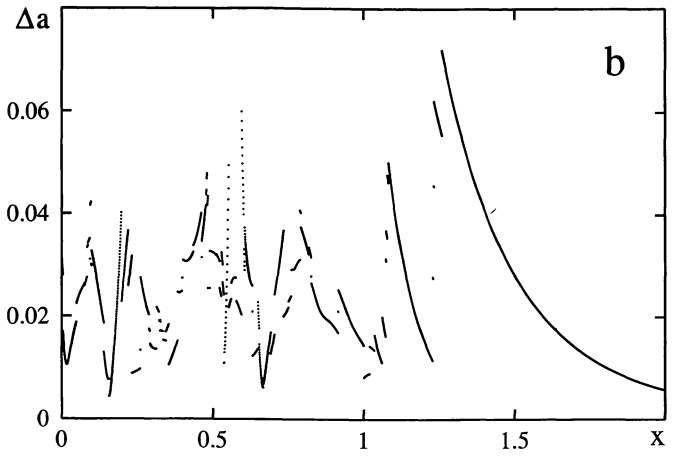
b) Quantitative Tests of Force Calculation

The qualitative picture presented above is supported by numerical experiments. For example, we generated a uniform-density sphere of $N = 4096$ particles with unit radius and measured the errors in the potential $\Delta\Phi \equiv \Phi_t - \Phi$, and acceleration $\Delta a \equiv |\boldsymbol{a}_t - \boldsymbol{a}|$ at 4096 points uniformly spaced on the x-axis between $x = 0$ and $x = 2$. Results computed for $\theta = 1$ and $\varepsilon = 0$ are plotted in Figure 9. The "fractal sawtooth" pattern of error as a function of x reflects the changing set of interactions used by the hierarchical algorithm to approximate the gravitational potential and force; it is a one-dimensional shadow of the tree itself. Major discontinuities represent wholesale revisions due to changes with respect to distant massive cells, while smaller jumps reflect local shifts. Note the bunching up of similar discontinuities, for example at $r \simeq 1.25$, directly illustrating the correlated errors produced by adjacent cells.

The distribution of $\Delta\Phi$ values also provides evidence for correlated errors. Figure 10 presents histograms of $\Delta\Phi$ for







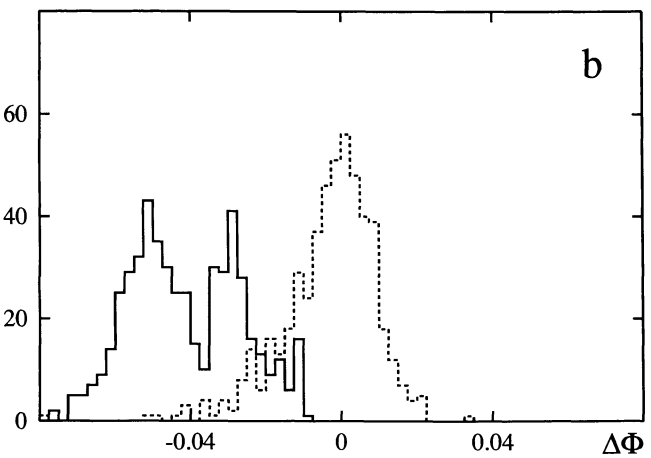


FIG. 9. FIG. 10.

FIG. 9.—(*a*) Error in potential $\Delta\Phi$ and (*b*) error in acceleration Δa plotted vs. position r for a system of $N = 4096$ particles in a homogeneous sphere of radius 1, sampled at $n = 4096$ points along the x-axis. The force calculation tolerance was $\theta = 1$, quadrupole moments were not included, and $\epsilon = 0$.

FIG. 10.—Distribution of $\Delta\Phi$ values for four different configurations, sampled at $n = 512$ interior points. Force calculation parameters follow Fig. 9. (*a*) "Ball" (*solid line*) and "King" (*dotted line*). (*b*) "Disk" (*solid line*) and "hierarchy" (*dotted line*).

four different mass distributions. Each configuration is represented by $N = 4096$ massive bodies and tested at $n = 512$ field points drawn from the same distribution; further details of these tests are given in Appendix D. These histograms are not Gaussian, nor are they centered on $\Delta\Phi = 0$, as we would expect if the errors due to individual cells add incoherently and the total error is not dominated by any particular cell.

Figure 11 shows the rms relative local error in the potential, $\langle(\Delta\Phi/\Phi)^2\rangle^{1/2}$, as a function of opening angle θ, for our four different mass configurations. When quadrupole moments are included, the errors are relatively insensitive to the form of the mass distribution, ranging from $\sim 10^{-3}$ for $\theta = 1$ to $\sim 10^{-5}$ for $\theta = 0.3$. Quadrupole corrections are particularly useful when calculating the forces due to the highly flattened "disk" distribution, which is not surprising since in this case many cells must have large quarupole moments. The "King" distribution appears to generate the smallest quadrupole corrections. The rms error converges rapidly as $\theta \to 0$, scaling roughly as θ^2 when forces are calculated without quadrupole corrections and as θ^4 when these are included.

In practice, we need to know not the error in the potential, but the error in the acceleration. Root-mean-square relative errors in the computed acceleration, $\langle|\Delta\boldsymbol{a}|^2/|\boldsymbol{a}|^2\rangle^{1/2}$, are shown in Figure 12. For a given configuration and θ, the rms error in $\boldsymbol{a}$ is roughly an order of magnitude larger than the rms error in Φ. These errors are also somewhat more sensitive to the configuration used. The slopes of these curves and the overall pattern of errors are very similar to Figure 11.

As noted in § I*a*, the tree algorithm yields a slightly biased result for the net potential energy U of a configuration. Figure 13 shows $|\langle\Delta\Phi\rangle/\langle\Phi\rangle|$, the fractional error in U, for the same set of configurations and θ values used above. Note that the average offset, $\langle\Delta\Phi\rangle$, may have either sign, although this information has been suppressed. While the measured bias is quite sensitive to the details of the configuration tested, the parameters used for the force calculation, and even the placement of the system with respect to the root cell of the tree, the magnitude of this bias is relatively small and can be reduced much further by using quadrupole corrections.

The tree algorithm introduces additional errors when a nonzero softening parameter ϵ is used, as Hernquist (1987)

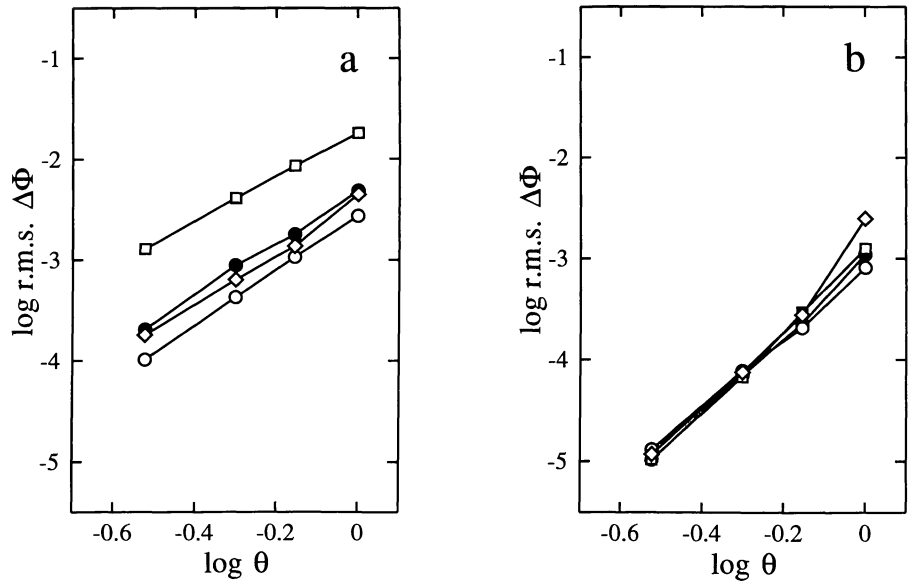


FIG. 11.—Root-mean-square relative errors in the potential, $\langle(\Delta\Phi/\Phi)^2\rangle^{1/2}$, averaged over $n = 512$ test positions. (*a*) Without quadrupole correction; (*b*) with quadrupole correction. Plotting symbols indicate the configuration tested: "ball" (*filled circles*), "King" (*open circles*), "disk" (*squares*), and "hierarchy" (*diamonds*).

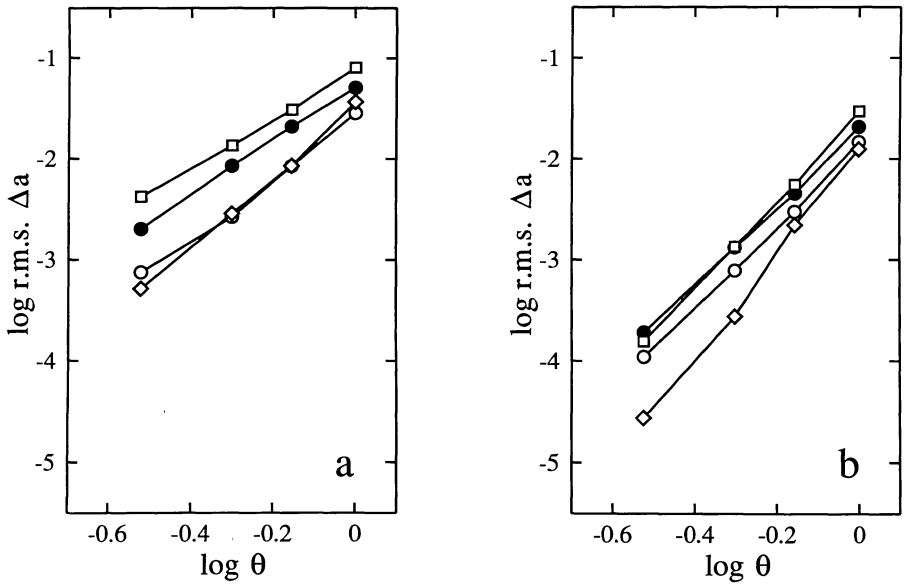


FIG. 12.—Root-mean-square relative errors in acceleration, $\langle|\Delta\boldsymbol{a}|^2/|\boldsymbol{a}|^2\rangle^{1/2}$. (*a*) Without quadrupole correction; (*b*) with quadrupole correction. Plotting symbols follow Fig. 11.

observed. In the $N \to \infty$ limit,[3] these additional errors come from cells within a distance of a few ε, and the contribution of such cells to the total error can hardly be greater than their contribution to the total force. Hence the tree-code errors due to $\varepsilon > 0$ will not dominate the errors already committed for zero softening unless ε is so large as to strongly perturb the global force field of the system. For finite N the effect of softening on the tree errors is even smaller, because some of the interactions calculated now involve particles, which are softened exactly. These considerations are supported by numerical results, presented in Figure 14, which shows the rms error in Φ as a function of ε for $\theta = 1$. Note that softening has no significant effect on the errors until ε is of the order of the characteristic scales of the system.

What is the cost of a force calculation? Figure 15 shows, for the same set of configurations and θ values as above, the mean number of total interactions, $\bar{n}_{\rm pn}$, and the mean number of particle-cell interactions, $\bar{n}_{\rm pc}$. These results, obtained with 4096 particles, scale roughly as log N for larger N. There is a significant dependence on the shape of the mass distribution,

[3]Note that in this limit, Hernquist's parameterization of ε in terms of the mean interparticle separation cannot be defined.

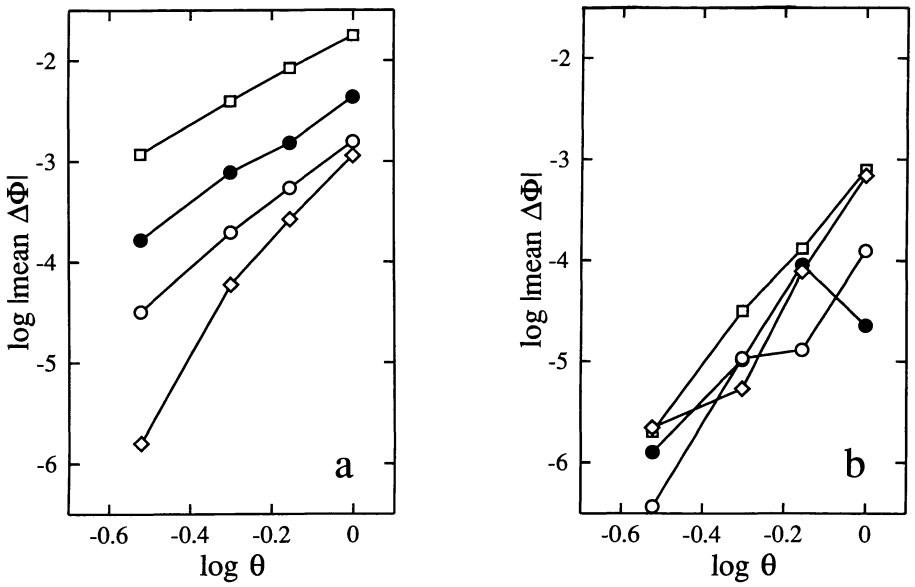


FIG. 13.—Relative offset of the potential, $|\langle\Delta\Phi\rangle/\langle\Phi\rangle|$. (*a*) Without quadrupole correction; (*b*) with quadrupole correction. Plotting symbols follow Fig. 11.

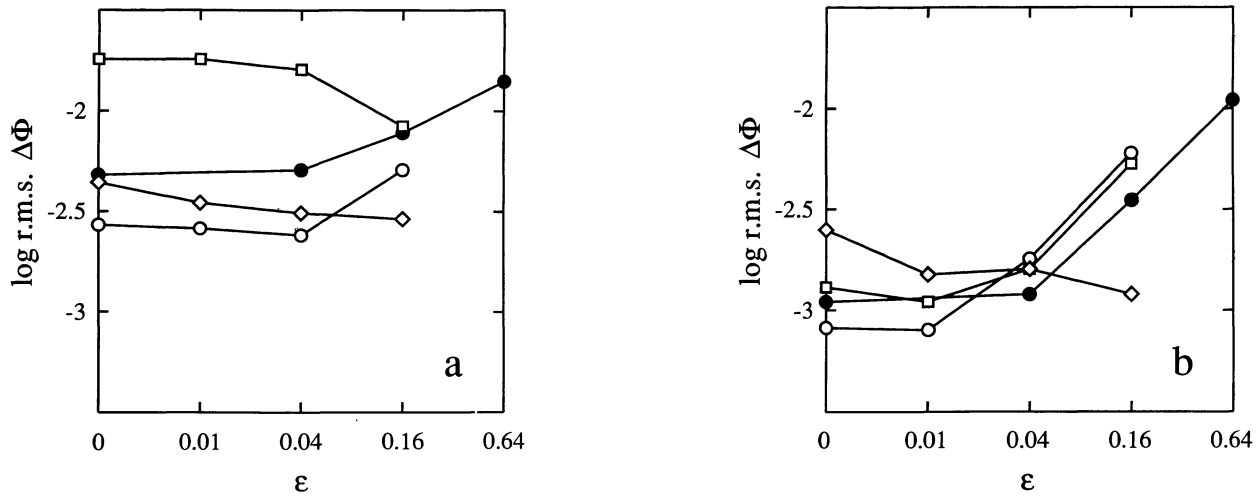


FIG. 14.—Root-mean-square relative errors in the potential, $\langle(\Delta\Phi/\Phi)^2\rangle^{1/2}$ as a function of ε, evaluated with opening angle $\theta = 1$. (*a*) Without quadrupole correction; (*b*) with quadrupole correction. Plotting symbols follow Fig. 11.

with "King" models requiring up to an order of magnitude more interactions per force calculation than hierarchical "cluster" configurations do. This is due to the extremely sparse structure of the tree derived from such a hierarchically clustered distribution. Flattened configurations are also relatively cheap, since again many of the cells examined are completely empty.

The total computational cost depends on the computer used, but scales roughly like $\bar{n}_{\rm pn} + Q\bar{n}_{\rm pc}$, where Q is the relative cost of a quadrupole correction, ranging from $Q \simeq 1$ on a SUN 3/50 with f68881 chip to $Q \simeq 0.3$ on a typical supercomputer; $Q = 0$ if quadrupole corrections are not used. In general, it proves to be efficient to use quadrupole moment even when only moderately accurate forces are required. For example, for an rms error of less than $\sim 3\times10^{-4}$ in $\Delta\Phi/\Phi$, we can set $\theta = 0.7$ and use quadrupole corrections, or set $\theta = 0.3$ and do without. In terms of total cost for comparable accuracy, the quadrupole option is anything from 1.2 to 4.0 times faster, depending on the machine and mass distribution involved.

When compared with direct-sum methods, the efficiency of a tree code depends on the total N, the force calculation accuracy required, the shape of the mass distribution, and the kind of hardware used. A direct-sum code must evaluate $(1/2)(N-1)$ interactions per particle per force calculation, in comparison with the effective $\bar{n}_{\rm pn} + Q\bar{n}_{\rm pc}$ evaluated by our tree code. Much in the spirit of computer hardware vendors, we cite the ratio $(1/2)(N-1)/(\bar{n}_{\rm pn} + Q\bar{n}_{\rm pc})$ as the smallest

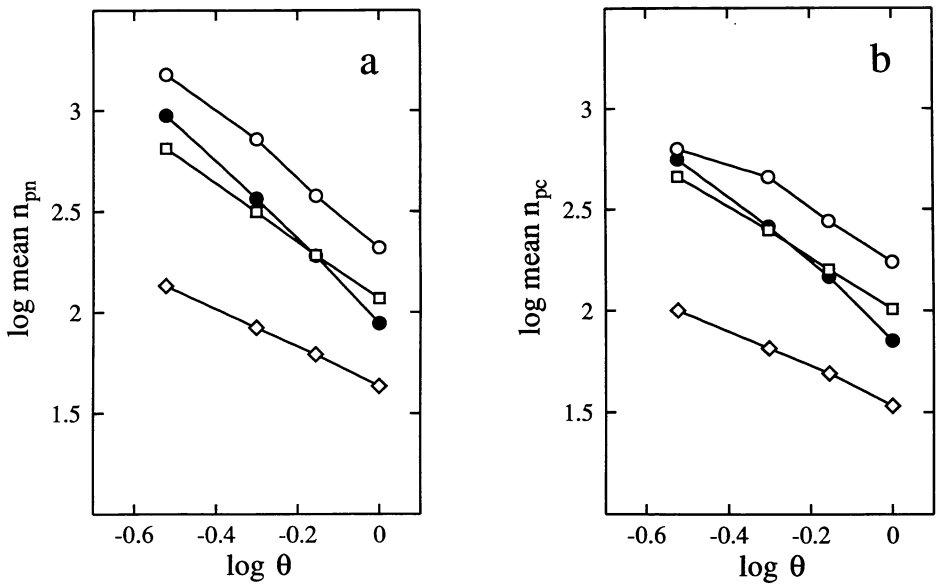


FIG. 15.—Average number of interactions calculated per particle per force calculation. (*a*) Mean number of total interactions, $\bar{n}_{pn}$; (*b*) mean number of particle-cell interactions, $\bar{n}_{pc}$. Plotting symbols follow Fig. 11.

speedup, relative to a direct-sum algorithm, that the tree code is "guaranteed not to exceed" (Dongarra 1988). This comparison, however, ignores the extra overhead of the recursive descent process, which can be very significant if the hardware is not hospitable. Perhaps a more useful figure is the N at which the tree code breaks even with a direct-sum method. For a low-accuracy calculation running on a workstation or mainframe, this point is reached by $N \sim 10^{2.5}$, while for production-quality runs on a vector supercomputer, break-even can require $N \sim 10^3$–$10^{3.5}$, because of the difficulty of mating the algorithm to the hardware.

IV. ERROR GROWTH IN DYNAMICAL N-BODY CALCULATIONS

How does the "noise" injected by hierarchical force calculations effect the evolution of an N-body system? A relative error ε in the force should lead to a relative error δ of order ε in the phase-space coordinates of each particle after one dynamical time. However, force calculation errors should be compared with other errors in N-body simulations: truncation in integrating the equations of motion, roundoff due to finite wordlength, force softening used to render the equations nonsingular, finite-N sampling of collisionless systems, and regularization transformations and perturbation approximations introduced for collisional systems. Given the scope for complicated interactions between various sources of error, a theoretical approach again seems rather daunting, but as in the last section, numerical experiments can be used to draw some definite conclusions, at least for collisionless systems. It is much harder to rigorously verify the results of collisional calculations, a point we will come back to.

To study error effects in an N-body calculation, we have run a sequence of models starting from the same initial conditions each time but changing parameters such as the time step Δt, softening length ε, total particle number N, and of course the opening angle θ. For these test calculations we chose the same collision of King models used in § I*a*, since problems of this general type highlight the versatility of tree codes. We evaluated the level of agreement between two calculations by comparing the coordinates of corresponding particles. Good consistency at this detailed level seems essential if projects such as the classification of orbits in merger remnants are attempted.

For the first set of experiments, we fixed the particle number at $N = 4096$ and the softening parameter at $\varepsilon = 1/40$. We varied the time step Δt from 1/40 to 1/160 and force calculation accuracy from $\theta = 1$ without quadrupole moments to $\theta = 0.5$ with them. Let $\mathbf{r}_i(t; a)$ be the position vector of particle i at time t in simulation a. At $t = 4$, when the merger is almost over, we compared simulations in pairs a, b by computing distances $\Delta r_i \equiv |\mathbf{r}_i(4; a) - \mathbf{r}_i(4; b)|$ for particles $i = 1, \ldots, 4096$. Figure 16 shows the grid of models; the numbers are median values of Δr_i evaluated between each adjacent pair of simulations. As the time step and force calculation are refined, median discrepancies between simulations tend toward zero; this is precisely the behavior we want the code to exhibit. It is also interesting that the errors due to the force calculation and integration algorithms are weakly coupled: the convergence on refining Δt by a factor of 2 is more rapid for smaller θ values, suggesting that force calculation errors are giving the leapfrog integrator a little indigestion. In the lower-right corner of Figure 16, the agreement between simulations is good enough that similar "constellations" of particles are apparent in the envelopes of the merger remnants, although such one-to-one correspondence cannot be found looking toward the centers of these systems.

For the next set of tests, we took the same initial conditions, used $\theta = 0.7$ with quadrupole moments, and varied the time step Δt from 1/80 to 1/320 and softening parameter ε from 1/40 to 1/160. The models and resulting median phase-space distances are shown in Figure 17. The discrepan-

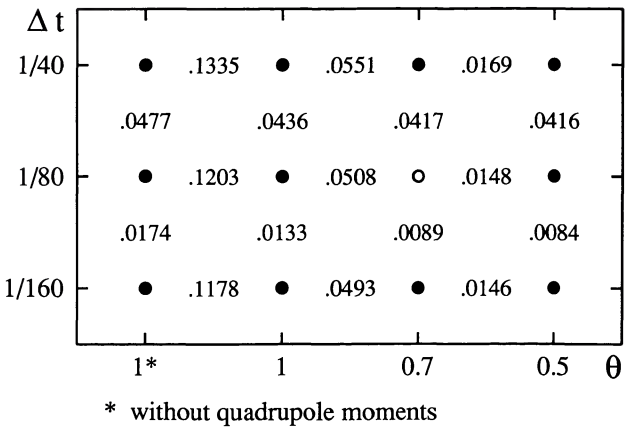


FIG. 16.—Grid of models in time step vs. force calculation accuracy plane, all starting from the same initial conditions; each dot represents a calculation. The numbers plotted between models are median distances for corresponding particles. The left-most column was made without quadrupole corrections; all others included them. The run marked with an open circle is common to the next two figures as well.

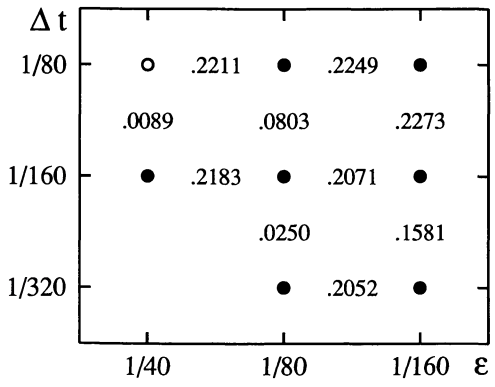


FIG. 17.—Grid of models in time step vs. softening length plane, all starting from the same initial conditions; each dot represents a calculation. The numbers plotted between models are median distances between corresponding particles.

cies between these simulations do *not* decrease as Δt and ε are jointly refined. For a given ε the models converge as Δt is reduced, but the converse is false. Furthermore, these two parameters are strongly coupled; a smaller ε demands a smaller Δt to reach a given accuracy.

The large effect seen here on changing ε requires an explanation; since the softening lengths used are an order of magnitude or more smaller than the core radii of the model galaxies, it seems unlikely that global differences in the potential are wholely responsible. The ever more lumpy potential revealed as ε is reduced increases the rate of two-body relaxation, deflecting particle trajectories. Note that ε is some two orders of magnitude greater than the 90° deflection distance; this means that softening directly sets the lower cutoff of the Coulomb integral. For the parameters used here, each particle has a few encounters within radius ε per crossing time. The resulting deflections are small, $\sim 10^{-2}$ rad, but after several crossing times even small deflections can become significant. Furthermore, the $1/(r^2+\varepsilon^2)^{1/2}$ softened potential used here departs significantly from $1/r$ out to several ε, so even wider encounters and the detailed response to small-scale structural features such as the cores of the two galaxies may be sensitive to changes in ε. This interpretation is in accord with the conventional wisdom that the most accurate solution to the original collisionless problem is obtained not with the "unphysical" parameter ε set to zero, but instead with some compromise value providing as much smoothing as possible without misrepresenting the continuum potential.

In order to test this interpretation directly, we ran a set of calculations fixing $\Delta t = 1/80$ and $\theta = 0.7$ with quadrupole moments, while varying the particle number N from 2048 to 16,384, and softening parameter ε from 1/20 to 1/80. To compare these simulations on a particle-by-particle basis we must precisely define the corresponding particles. Given a system of N particles, we can sample one out of every two to get a similar system with only $N/2$ particles. If the discreteness of an N-body system had no effect on the initial specification or subsequent evolution, this sampling would commute with time integration; that is, it would not matter if we sampled first and integrated after, or the other way around. We have therefore iterated the sampling operation to obtain the set of initial conditions used for Figure 18 and have derived the median particle displacements at $t = 4$ by comparing the coordinates of those particles retained by sampling. As N increases, the discrepancy between these models tends to vanish, roughly consistent with the relaxation rate scaling like $N^{-1/2}$ as expected. The relaxation rate increases with decreasing ε, and the median discrepancy between $\varepsilon = 1/40$ and $\varepsilon = 1/80$ decreases with increasing N, as also expected. However, the models with $\varepsilon = 1/20$ are severely affected by excessive softening, which perturbs the net binding energies by $\sim 5\%$ and leads to the large differences, independent of N, with the $\varepsilon = 1/40$ models.

A comparison of Figures 16 and 18 shows that, for the most part, numerical errors due to finite θ and Δt are smaller than the effects due to finite N. It is clear that the bulk of this relaxation is "physical," given the actual N of the model. For example, consider the common run marked with an open circle, which appears to be an order of magnitude closer to the $\theta \to 0, \Delta t \to 0$ limit than to the $N \to \infty$ limit, as inferred

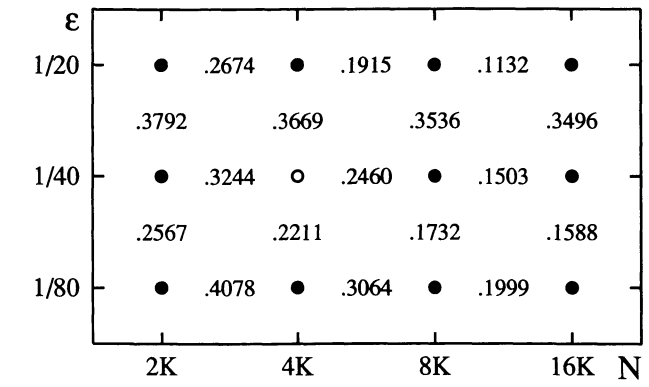


FIG. 18.—Grid of models in softening length vs. particle number plane, all starting from similar initial conditions; each dot represents a calculation. As described in the text, a sampling procedure is used to obtain a subset of particles common to all four models. The numbers plotted between models are median phase space distances for these particles. Note that 1K ≡ 1024.

from the median Δr_i values. We conclude that tree methods do not suffer from unfavorable levels of two-body relaxation; this result is consistent with test-particle experiments described by Hernquist (1987). Further improvement in tree methods for a fixed N is much less desirable than simply increasing N for the tree codes which are currently available.

In hindsight it is not too surprising that relaxation rates in tree codes should be quite comparable to those of direct-summation codes using the same value of ε. One might worry that since each particle in a tree code effectively "feels" only $n_{pn} \ll N$ other objects, the relaxation rate should be comparable to that found in an N-body system of $\sim n_{pn}$ particles, regardless of the actual value of N. The answer to this worry is that the distant, massive cells are distributed in a rather regular manner, with masses smoothly varying in time, and are certainly not subject to the level of Poisson fluctuation expected for n_{pn} independent points.

Note that it is not clear from the above calculations to what extent tree methods reproduce the evolution of collisional systems correctly; the evolution of a globular cluster depends on a very fine balance of competing effects, which have not been explored in the above tests. All we can say from our results is that, in principle, tree methods can be used for globular cluster simulations, since the errors rigorously approach zero for $\theta \to 0, \Delta t \to 0$. Whether such calculations would be cost effective is a different question, discussed by Hut, Makino, and McMillan (1988).

V. CONCLUSIONS

We have investigated the error characteristics of a hierarchical N-body algorithm for computing the dynamics of a self-gravitating particle system. Our algorithm is based on recursively refining regions of space to identify groups of particles whose gravitational field can be approximated by a low-order multipole expansion (Barnes and Hut 1986). In our method the tree is of Eulerian type, and the connections with the particles are found through the recursive refinement. Another Eulerian approach, with a different method of force expansions, was developed by Greengard and Rokhlin (1987).

Earlier tree algorithms tried to identify natural clumping structures within the particle distribution in an effort to find an optimal association between tree nodes and particle clumps (Appel 1981, 1985; Jernigan 1985; Porter 1985). Thus, these codes have a Lagrangian character. These algorithms also used the tree structure to expedite the integration of the equations of motion, as well as the process of force calculation, a bold stroke which may not have paid off in practice. The most recent Lagrangian tree method, mentioned by Press (1986) and further described by Benz *et al.* (1989), conforms to our approach in using the tree for force calculation only.

The use of tree methods in stellar dynamics is still sufficiently novel that it is too early to say which methods are most efficient for which type of applications. Since we have more experience with Eulerian codes, let us mention here some of the aspects of Eulerian trees in which we consider them to be superior to Lagrangian trees. One immediate advantage of an Eulerian tree is the *simplicity of the tree construction*. As a corollary, another advantage is the possibility of a *rigorous error analysis*, since upper bounds can be given for the higher order multipole moments because of the spatially regular cubical structure of the branches of the tree. Indeed, most of the present paper presents a systematic exploration of the behavior of the errors in our code in a variety of circumstances.

We began our investigation in § II by considering the local static error made when some complicated but distant set of particles is represented by a single point mass. We found analytical expressions for these errors in a variety of cases, as derived in Appendices B and C. For a homogeneous particle distribution in a box, the leading non-Keplerian term in a potential expansion is its hexadecapole moment ($l = 4$) in the limit of a smooth particle distribution, i.e., for an infinitely large number of particles. For a distribution with a linear density gradient, extra contributions arise starting at the quadrupole moment ($l = 2$). The analytic expressions were compared with the results of numerical experiments, confirming the analytic theory but also illustrating the complicated interplay of effects due to finite particle number. In particular, when a single cell contained less than 10^3–10^4 particles, random noise from the discreteness of the particle distribution would generate quadrupole and octopole ($l = 3$) moments at least as large as hexadecapole moments, even though the particles were drawn from an underlying homogeneous distribution with vanishing $l = 2$ and $l = 3$ multipole moments.

With a good understanding of the error due to a single cell, the next logical step was to consider the error in the total force and potential at a given place. While the total error is just the sum of the errors due to the individual cells in the interaction list, evaluation of this sum requires consideration of the geometry of the mass on large scales *and* the level of discreteness on small scales. Straightforward numerical experimentation is probably the only practical way to do the job in realistic cases. Results presented in § III show that the accuracy and computational cost of our tree algorithm depend on details of the mass distribution, with hierarchical structures, of course, eliciting the best overall performance. In all cases, the level of error can be controlled at a reasonable cost by

careful choice of the opening angle θ. Much of the dependence on configuration appears to be due to correlations present in the errors from different cells; shapes with large-scale features extending over many cells, such as the "ball" and "disk," provide the most impressive evidence for correlated error patterns. The hierarchical force-calculation algorithm becomes competitive with direct-sum methods for $N \sim 10^{2.5}$ to $N \sim 10^{3.5}$, depending on the hardware, mass configuration, and required precision.

Once the accuracy of force calculations has become, via θ, an adjustable parameter in an N-body calculation, one must know how to pick an optimum value. We define this optimum in terms of a measure of the distance between two N-body systems derived from the displacements of individual particles. In § IV we present a series of realistic, full-scale numerical experiments using a head-on merger of two spherical galaxies to contrast the effects of finite force-calculation errors with other well-nigh universal sins of N-body simulations: integration errors, force softening, and Monte-Carlo sampling of continuous systems. Figures 16–18 show how these error sources compete with each other; over most of the parameter space considered here, the dominant source of error, often by an order of magnitude, is the discrete representation of the ideal continuum problem originally posed. Note that this result could not have been obtained by examining globally conserved quantities such as E and $\boldsymbol{J}$, since these are conserved even in collisional systems, while the detailed phase-space structure is not. Right off the bat, we can conclude that tree methods do not *amplify* the intrinsic discreteness of N-body systems, for if they did, the effect of using $\theta > 0$ would dominate the effect of using $N < \infty$. Note however that, while tree codes do not exhibit a higher *level* of relaxation than direct-sum methods do, we have not yet investigated whether tree methods will get the *details* of collisional systems right.

A separate question, discussed by Hernquist and Barnes (1989) is whether, *for a given level of spatial resolution*, direct-sum and tree methods suffer more from relaxation effects than so-called collisionless methods based on Fourier or spherical harmonic expansions. From theoretical considerations, together with a small number of N-body, experiments, Hernquist and Barnes conclude that the relaxation rate depends on the *resolution* of the force calculation, but not on the details of the algorithm used. Thus we are free to pick the most efficient code capable of delivering the resolution required for the problem at hand. A more detailed comparison of tree codes with other methods, especially potential-expansion codes, would be very welcome, now that the theoretical and experimental understanding of N-body methods is developing some maturity. It would be especially instructive to define a suite of standard N-body initial conditions which could be fed to different codes in order to make a detailed comparison in terms of cost and efficiency.

Finally, we contrast our algorithm with the $O(N)$ method of Greengard and Rokhlin (1987). There are considerable similarities in the two methods; as Katzenelson (1989) has shown, both can be viewed as solving a certain recurrence relation. The main difference lies in the representation used: the $O(N)$ method constructs an approximation to the *field* itself, $\Phi_C(\boldsymbol{r})$, within each cell C, while our $O(N \log N)$ method is formulated in terms of *interactions* between particles. Indeed, it is the funneling of information from the $O(\log N)$ levels of the tree into Φ_C, an object with fixed complexity, which gives the Greengard and Rokhlin method its theoretical $O(\log N)$ advantage. Such empirical comparisons as are available, however, suggest that the $O(N)$ method does not have a great advantage for the modest levels of accuracy required in present astrophysical applications. The $O(N)$ method is also more difficult to program, especially in the form required for highly inhomogeneous systems (Carrier, Greengard, and Rokhlin 1988). Short-term practical considerations aside, the Greengard and Rokhlin method does appear to represent a breakthrough in principle.

We conclude that presently implemented tree-codes can indeed provide reliable numerical simulations of collisionless systems, since there is now a well-understood procedure to obtain arbitrary accuracy in an efficient and controlled way. Having sharpened our arboreal tools, we are now in a position to apply them with confidence to a large variety of problems, which have in common merely a complexity in geometry and large deviations from homogeneity.

We thank Lars Hernquist, Gerald Sussman, and Martin Weinberg for interesting discussions and Mathew Halfant for writing the special-purpose symbolic manipulation routines used to compute the formulae in Appendices B and C. Numerical calculations were run at the John von Neumann Center under LAC-17016. J. B. was supported by New Jersey High Technology Grant No. 88-240090-2.

APPENDIX A

THE BARNES-HUT TREE ALGORITHM

The algorithm used for the tests reported here was informally described by Barnes and Hut (1986). Here we give a more detailed description, based on a revised implementation. Copies of our revised code in C and Fortran 77 are available on request.

The tree structure used by our code is represented by a collection of *cells*, which correspond to internal nodes of the tree, and *particles*, which correspond to the leaves at the edge of the tree. Each cell holds the addresses of up to $2^3 = 8$ *descendents*, which may be either cells or particles. The data structures are tied together using pointers to form a directed graph, which may be followed from the *root* cell out toward the leaves. Particles and cells have several components in common, including a *mass* and a *position vector*. Cells may also have a *quadrupole moment* for the mass enclosed, if this feature is enabled.

At each time-step during an N-body calculation, before forces can be calculated, the tree must be constructed anew from the coordinates of the particles. This is done by first creating an empty tree, and then inserting the particles into it one at a time. To

insert a particle, compute its coordinates (x, y, z) scaled to the range [0,1) within the root cell. Consider a binary expansion of these scaled coordinates:

$$x = 0.x_1x_2x_3\ldots, \tag{A1}$$

where each x_i is either 0 or 1, and likewise for y and z. The top bits of these three coordinates, x_1, y_1, and z_1, indicate which of the eight descendents of the root this particle should go in.[4] If the descendent slot is occupied by another cell, the algorithm moves down a level and looks at the next three bits, x_2, y_2, and z_2. If the slot is occupied by a particle, a new cell is inserted in the slot, and both the old and new particles are then inserted in the new cell. If there is nothing in the descendent slot, we simply insert the particle. Thus, starting from the root node, the insertion algorithm scans down the tree looking for a place to insert the current particle and extending the tree as necessary whenever two particles try to occupy the same slot. Tree construction is concluded by computing the aggregated parameters, namely the mass, center-of-mass position vector, and optional quadrupole moment, of each cell. This computation is performed by making a single recursive traverse of the entire tree structure, visiting all the nodes below any particular cell before computing parameters for the cell itself.

A few comments about the implementation of the construction algorithm: (1) In our current version, the root cell is dynamically expanded during tree construction if a particle which does not fit inside is found; the new root cell is chosen to contain the old root as a subcell, permitting us to "repot" the partly constructed tree. It would have been simpler to make a preliminary pass through the particle coordinates and fit the root cell around them beforehand! (2) Scanning down the tree when inserting a particle can be easily accomplished by a simple iterative procedure; the recursive tree construction routine found in early versions of our tree code has now been replaced. (3) In the F77 version, aggregated parameters are computed by listing the cells in order of increasing size and processing iteratively in this order; this ensures that aggregated parameters from level k are available to $k-1$. (4) In one parallel implementation (Barnes 1986), the tree is constructed by first sorting the particles by the value obtained by shuffling the bits of (x, y, z) together as

$$(0.x_1x_2x_3\ldots, 0.y_1y_2y_3\ldots, 0.z_1z_2z_3\ldots) \rightarrow 0.x_1y_1z_1x_2y_2z_2x_3y_3z_3\ldots, \tag{A2}$$

and then grouping particles into cells from the bottom up. This bit-shuffling operation is a nice application of Cantor's one-to-one map between the unit segment [0,1) and the unit square $[0,1)^2$.

To approximate the force on a particle p, the algorithm makes a recursive traverse of the tree structure, starting at the root cell and exploring different parts of the tree at different levels of resolution. Suppose the algorithm is currently working on node q of the tree; there are three possibilities. (1) If q is a particle, calculate the interaction between p and q. (2) If q is a cell for which *can_accept*$(q, p; \theta)$ is true, calculate the interaction between p and q, optionally including a quadrupole correction for q. (3) If q is a cell for which *can_accept* is false, do not calculate the interaction between p and q directly, but examine the subcells of q according to the above rules instead. The predicate *can_accept*$(p, q; \theta)$ quickly guesses if q is sufficiently well separated from p to be handled with a single interaction. Currently, we use a simple and purely geometrical condition; let d_{pq} be the distance from p to the center of mass of q, and l_q be the length of one side of q; then

$$\textit{can\_accept}(q, p; \theta) \equiv l_q < \theta d_{pq}. \tag{A3}$$

Some comments about the tree-walk algorithm: (1) The recursive procedure outlined above may be recast in an iterative form by keeping a stack of nodes to examine: push the root node onto the stack, then loop until the stack is empty, popping the top node q off and either calculating its interaction with p or pushing its descendents on the stack instead. (2) For $\theta \simeq 1$ the predicate *can_accept* may occasionally fail to detect that p is actually *inside* the cell q, introducing a potentially serious error in the computed force. In practice this happens so rarely that it is sufficient to flag such errors when they occur. A more rigorous solution would be for *can_accept* to explicitly check if p lies within q, but the improvement is not worth the additional computation. (3) The problem of vectorizing a tree-walk algorithm is not trivial, because a recursive flow of control is at odds with the linear organization of vector processors. For various approaches to vectorization, see Makino (1989), Hernquist (1989), and Barnes (1989).

In the standard version of the tree code we use a leapfrog integrator to advance the coordinates of the particles. As a simple low-order scheme which does not require significant storage, a leapfrog seems well adapted to the kinds of calculations so far run with our code. One drawback of the leapfrog is that positions and velocities are a half-step out of phase with each other. In order to start the calculation or extract a synchronized phase-space snapshot, we need an auxiliary formula. Since leapfrog integrators make global errors of $O(\Delta t^2)$, the starting/stopping formulae need not be more accurate than $O(\Delta t^2)$ locally. We therefore use

[4]Note that *any* one-to-one mapping between the three bits x_i, y_i, z_i and the descendent index will serve, but to simplify debugging a map such as $x_i, y_i, z_i \rightarrow 4x_i + 2y_i + z_i$ is preferred.

the following scheme:

$$\begin{aligned} v_{1/2} &= v_0 + (1/2)\,\Delta t a(x_0), \\ x_1 &= x_0 + \Delta t v_{1/2}, \\ v_{3/2} &= v_{1/2} + \Delta t a(x_1), \\ &\cdots \\ v_{n-1/2} &= v_{n-3/2} + \Delta t a(x_{n-1}), \\ x_n &= x_{n-1} + \Delta t v_{n-1/2}, \\ v_n &= v_{n-1/2} + (1/2)\,\Delta t a(x_n). \end{aligned} \tag{A4}$$

This formulation is manifestly time-reversible. An improved scheme which corrects the start-up error is described by Hernquist and Katz (1989).

APPENDIX B
THE LARGE-N LIMIT

Here we derive an expression for the gravitational potential energy $\Phi(\boldsymbol{r})$ outside a cubic cell with constant density ρ_0, accurate to seventh order in a/r with a the length of the side of the cube, centered on the origin of a Cartesian coordinate system, and $r = |\boldsymbol{r}|$ the distance from the center of the box to the point of measurement of the potential. Since it is hard to find recipes for this type of calculation in modern textbooks (in contrast to older works such as Kellogg [1929], chap. V), we felt that it would be useful to present a self-contained derivation below, even though this implies a duplication of those aspects of the calculation which are easy to find in textbooks.

The cubic mass distribution,

$$p(\tilde{x}, \tilde{y}, \tilde{z}) = \begin{cases} \rho_0, & |\tilde{x}| < \frac{1}{2}a,\ |\tilde{y}| < \frac{1}{2}a, \text{ and } |\tilde{z}| < \frac{1}{2}a, \\ 0, & \text{otherwise,} \end{cases} \tag{B1}$$

implies a total mass

$$M = \rho_0 a^3. \tag{B2}$$

We will use units in which the gravitational constant $G = 1$. In our notation $\tilde{\boldsymbol{r}} = (\tilde{x}, \tilde{y}, \tilde{z})$ indicates the position of a mass element $\rho_0\, d\tilde{x}\, d\tilde{y}\, d\tilde{z}$ inside the cube and $\boldsymbol{r} = (x, y, z)$ indicates any point outside a sphere enclosing the cube, i.e., $x^2 + y^2 + z^2 > (3/4)a^2$.

The potential

$$\Phi(\boldsymbol{r}) = \rho_0 \int_{-a/2}^{a/2} d\tilde{x} \int_{-a/2}^{a/2} d\tilde{y} \int_{-a/2}^{a/2} d\tilde{z}\, \frac{1}{|\boldsymbol{r} - \tilde{\boldsymbol{r}}|} \tag{B3}$$

can be approximated by the usual expansion of the inverse distance between $\boldsymbol{r}$ and $\tilde{\boldsymbol{r}}$ in terms of the Legendre polynomials $P_l(u)$, (see Jackson 1975), where

$$u = \cos\gamma = \frac{\boldsymbol{r} \cdot \tilde{\boldsymbol{r}}}{r\tilde{r}} = \frac{x\tilde{x} + y\tilde{y} + z\tilde{z}}{r\tilde{r}} \tag{B4}$$

and γ is the angle between the vectors $\boldsymbol{r}$ and $\tilde{\boldsymbol{r}}$. We will carry the expansion

$$\frac{1}{|\boldsymbol{r} - \tilde{\boldsymbol{r}}|} = \frac{1}{r}\left[P_0(u) + \frac{\tilde{r}}{r} P_1(u) + (\tilde{r}/r)^2 P_2(u) + \cdots \right] \tag{B5}$$

to order $l = 6$. For completeness we list the P_l needed:

$$
\begin{aligned}
P_0(u) &= 1, \\
P_1(u) &= u, \\
P_2(u) &= \tfrac{3}{2}\left(u^2 - \tfrac{1}{3}\right), \\
P_3(u) &= \tfrac{5}{2}\left(u^3 - \tfrac{3}{5}u\right), \\
P_4(u) &= \tfrac{35}{8}\left(u^4 - \tfrac{6}{7}u^2 + \tfrac{3}{35}\right), \\
P_5(u) &= \tfrac{63}{8}\left(u^5 - \tfrac{10}{9}u^3 + \tfrac{5}{21}u\right), \\
P_6(u) &= \tfrac{231}{16}\left(u^6 - \tfrac{15}{11}u^4 + \tfrac{5}{11}u^2 - \tfrac{5}{231}\right),
\end{aligned}
\tag{B6}
$$

which leads to the expansion of the inverse distance in Cartesian coordinates as

$$
\begin{aligned}
\frac{1}{|\mathbf{r}-\tilde{\mathbf{r}}|} = {} & \frac{1}{r} + \frac{\mathbf{r}\cdot\tilde{\mathbf{r}}}{r^3} + \frac{1}{r^5}\left[\frac{3}{2}(\mathbf{r}\cdot\tilde{\mathbf{r}})^2 - \frac{1}{2}r^2\tilde{r}^2\right] + \frac{1}{r^7}\left[\frac{5}{2}(\mathbf{r}\cdot\tilde{\mathbf{r}})^3 - \frac{3}{2}(\mathbf{r}\cdot\tilde{\mathbf{r}})r^2\tilde{r}^2\right] \\
& + \frac{1}{r^9}\left[\frac{35}{8}(\mathbf{r}\cdot\tilde{\mathbf{r}})^4 - \frac{15}{4}(\mathbf{r}\cdot\tilde{\mathbf{r}})^2 r^2\tilde{r}^2 + \frac{3}{8}r^4\tilde{r}^4\right] \\
& + \frac{1}{r^{11}}\left[\frac{63}{8}(\mathbf{r}\cdot\tilde{\mathbf{r}})^5 - \frac{35}{4}(\mathbf{r}\cdot\tilde{\mathbf{r}})^3 r^2\tilde{r}^2 + \frac{15}{8}(\mathbf{r}\cdot\tilde{\mathbf{r}})r^4\tilde{r}^4\right] \\
& + \frac{1}{r^{13}}\left[\frac{231}{16}(\mathbf{r}\cdot\tilde{\mathbf{r}})^6 - \frac{315}{16}(\mathbf{r}\cdot\tilde{\mathbf{r}})^4 r^2\tilde{r}^2 + \frac{105}{16}(\mathbf{r}\cdot\tilde{\mathbf{r}})^2 r^4\tilde{r}^4 - \frac{5}{16}r^6\tilde{r}^6\right].
\end{aligned}
\tag{B7}
$$

We can now evaluate equation (B3) term by term in expansion (B7) by writing

$$
\Phi(\mathbf{r}) = \sum_{i=0}^{\infty} \Phi_i(\mathbf{r}), \tag{B8}
$$

with $\Phi_i(\mathbf{r})$ corresponding to the terms containing $P_i(u)$ in equation (B5):

$$
\Phi_i(\mathbf{r}) = \frac{\rho_0}{r^{2i+1}} \int_{-a/2}^{a/2} d\tilde{x} \int_{-a/2}^{a/2} d\tilde{y} \int_{-a/2}^{a/2} d\tilde{z}\, Q_i(\mathbf{r}, \tilde{\mathbf{r}}), \tag{B9}
$$

where $Q_i(x, y, z, \tilde{x}, \tilde{y}, \tilde{z})$ is a polynomial in which each term is of the form

$$
\tilde{x}^{\alpha}\tilde{y}^{\beta}\tilde{z}^{\gamma}x^{\kappa}y^{\lambda}z^{\mu}, \tag{B10a}
$$

with the constraint that the six nonnegative integers satisfy the relations

$$
\alpha + \beta + \gamma = \kappa + \lambda + \mu = i. \tag{B10b}
$$

Starting with the monopole term

$$
Q_0 = 1 \tag{B11}
$$

we recover the Keplerian potential which is the dominant part of the potential when $r \gg a$:

$$
\Phi_0 = \rho_0 \frac{a^3}{r} = \frac{M}{r}. \tag{B12}
$$

The dipole term

$$Q_1 = \mathbf{r}\cdot\tilde{\mathbf{r}} = x\tilde{x} + y\tilde{y} + z\tilde{z}, \tag{B13}$$

when inserted as the integrand in equation (B9) gives

$$\Phi_1 = 0 \tag{B14}$$

by symmetry, since the integration intervals are even in $\tilde{x}$ (as well as in $\tilde{y}$ and $\tilde{z}$), while Q_1 is odd in $\tilde{x}$ (as well as in $\tilde{y}$ and $\tilde{z}$). This cancellation argument extends to any Q_i with i odd, as follows directly from equation (B10) for which the reflection $\tilde{\mathbf{r}} \rightarrow -\tilde{\mathbf{r}}$ gives

$$Q_i(\mathbf{r}, -\tilde{\mathbf{r}}) = (-1)^i Q_i(\mathbf{r}, \tilde{\mathbf{r}}). \tag{B15}$$

For our present seventh-order expansion this implies:

$$\Phi_3 = \Phi_5 = \Phi_7 = 0. \tag{B16}$$

The quadrupole term

$$Q_2 = \tfrac{3}{2}(\mathbf{r}\cdot\tilde{\mathbf{r}})^2 - \tfrac{1}{2}(\mathbf{r}\cdot\mathbf{r})(\tilde{\mathbf{r}}\cdot\tilde{\mathbf{r}}) \tag{B17}$$

written out in the form of equation (B10) reads

$$Q_2 = \tilde{x}^2\left(x^2 - \tfrac{1}{2}y^2 - \tfrac{1}{2}z^2\right) + \tilde{y}^2\left(y^2 - \tfrac{1}{2}z^2 - \tfrac{1}{2}x^2\right) + \tilde{z}^2\left(z^2 - \tfrac{1}{2}x^2 - \tfrac{1}{2}y^2\right) + 2\tilde{x}\tilde{y}xy + 2\tilde{x}\tilde{z}xz + 2\tilde{y}\tilde{z}yz. \tag{B18}$$

Integration of Q_2 in equation (B9) gives

$$\Phi_2 = 0 \tag{B19}$$

even though each of the first three terms in equation (B18) does give a nonzero contribution. This cancellation is exceptional for Q_i of even i, and in fact all other $Q_i \neq 0$ with $i \neq 2$ even.

The physical reason for the cancellation is that Q_2 is a measure for the flattening of a (nonnegative) mass-density distribution, which is nonzero for a rectangular cell in general, but vanishes for a cubic cell. Mathematically, Q_2 vanishes by virtue of the Cartesian quadrupole moment being a traceless tensor, which is related to the redundancy in the degrees of freedom of a Cartesian multipole tensor as compared with an expansion in spherical harmonics (see Jackson [1975], p. 139).

The Q_i, $i \geq 4$ and i even, takes the general form

$$Q_i(x, y, z, \tilde{x}, \tilde{y}, \tilde{z}) = Q_i^+ + Q_i^-, \tag{B20}$$

with

$$Q_i^+ = \tilde{x}^{2\alpha}\tilde{y}^{2\beta}\tilde{z}^{2\gamma}x^{2\kappa}y^{2\lambda}z^{2\mu}, \tag{B21a}$$

with $\alpha, \beta, \gamma, \kappa, \lambda, \mu$ nonnegative integers satisfying

$$\alpha + \beta + \gamma = \kappa + \lambda + \mu = \frac{i}{2}, \tag{B21b}$$

while Q_i^- contains all other terms with odd powers in $\tilde{x}$, $\tilde{y}$, and $\tilde{z}$ (and in x, y, and z as well).

In the following analysis we are not interested in the Q_i^-, since they all integrate to zero in equation (B9), as did, for example, the last three terms in equation (B18).

We are now finally in a position to obtain the first nonvanishing correction term to the Keplerian approximation of the potential of a homogeneous cube, due to its nonvanishing hexadecapole moment (physically, a measure for symmetric "corneredness" of a distribution, nonzero for any rectangular mass, even a nonflattened cube).

With

$$Q_4 = \tfrac{35}{8}(\mathbf{r}\cdot\tilde{\mathbf{r}})^4 - \tfrac{15}{4}(\mathbf{r}\cdot\tilde{\mathbf{r}})^2(\mathbf{r}\cdot\mathbf{r})(\tilde{\mathbf{r}}\cdot\tilde{\mathbf{r}}) + \tfrac{3}{8}(\mathbf{r}\cdot\mathbf{r})^2(\tilde{\mathbf{r}}\cdot\tilde{\mathbf{r}})^2, \tag{B22}$$

writing out the dependence on Cartesian components and retaining even terms only gives

$$\begin{aligned} Q_4^+ &= \tilde{x}^4\left(\tfrac{35}{8}x^4 - \tfrac{15}{4}x^2r^2 + \tfrac{3}{8}r^4\right) + \tilde{y}^4\left(\tfrac{35}{8}y^4 - \tfrac{15}{4}y^2r^2 + \tfrac{3}{8}r^4\right) + \tilde{z}^4\left(\tfrac{35}{8}z^4 - \tfrac{15}{4}z^2r^2 + \tfrac{3}{8}r^4\right) \\ &+ \tilde{x}^2\tilde{y}^2\left(\tfrac{105}{4}x^2y^2 - \tfrac{15}{4}\left(x^2 + y^2\right)r^2 + \tfrac{3}{4}r^4\right) \\ &+ \tilde{x}^2\tilde{z}^2\left(\tfrac{105}{4}x^2z^2 - \tfrac{15}{4}\left(x^2 + z^2\right)r^2 + \tfrac{3}{4}r^4\right) + \tilde{y}^2\tilde{z}^2\left(\tfrac{105}{4}y^2z^2 - \tfrac{15}{4}\left(y^2 + z^2\right)r^2 + \tfrac{3}{4}r^4\right), \end{aligned} \tag{B23}$$

with $r^2 = x^2 + y^2 + z^2$. Substituting all this in equation (9) and rearranging terms finally gives

$$\Phi_4 = \frac{7}{960}\frac{Ma^4}{r^9}\left[3r^4 - 5\left(x^4 + y^4 + z^4\right)\right]. \tag{B24}$$

A direct check can be applied to this expression by computing the spherical average (keeping r fixed, but orienting $\boldsymbol{r}$ in arbitrary directions), which should vanish. This requirement holds true for any multipole moment beyond the monopole moment: adding the contribution felt by many observers at different orientation angles is equivalent to superposing the contributions of many cubes with different orientations at a fixed observer position. But in the limit of very many cubes, the superposed cubes form a spherical distribution with vanishing higher order moments beyond the monopole. Indeed,

$$\int_0^{\pi}\int_0^{2\pi}\left[\cos^4\theta + \sin^4\theta\left(\sin^4\phi + \cos^4\phi\right)\right]\sin\theta\, d\phi\, d\theta = \frac{12\pi}{5},$$

from which it follows that $\int\int \Phi_4 \sin\theta\, d\phi\, d\theta = 0$.

The recipe developed above can be extended arbitrarily. Here we terminate our expansion at the first nonleading term in the expansion of the non-Keplerian part of the potential of a cube, due to its 64-pole (hexacontatetrapole). For completeness we list Q_6^+ below, but in a form which is more economical than equation (B23). Let us denote by $\overline{Q}_i^+$ the expression which yields Q_i^+ by cyclicly permuting and adding terms:

$$Q_i^+(x, y, z, \tilde{x}, \tilde{y}, \tilde{z}) = \overline{Q}_i^+(x, y, z, \tilde{x}, \tilde{y}, \tilde{z}) + \overline{Q}_i^+(y, z, x, \tilde{y}, \tilde{z}, \tilde{x}) + \overline{Q}_i^+(z, x, y, \tilde{z}, \tilde{x}, \tilde{y}). \tag{B25}$$

For example, equation (B23) reduces to

$$\overline{Q}_4^+ = \tilde{x}^4\left(\tfrac{35}{8}x^4 - \tfrac{15}{4}x^2r^2 + \tfrac{3}{8}r^4\right) + \tilde{x}^2\tilde{y}^2\left[\tfrac{105}{4}x^2y^2 - \tfrac{15}{4}\left(x^2 + y^2\right)r^2 + \tfrac{3}{4}r^4\right]. \tag{B26}$$

Note that this reduction is far from unique; for example, we could equally well have taken from equation (B23) the 1st and 5th line instead.

In our new notation, the hexacontatetrapole contribution to the potential is generated by

$$\begin{aligned} \overline{Q}_6^+ &= \tfrac{1}{16}\tilde{x}^6(231x^6 - 315x^4r^2 + 105x^2r^4 - 5r^6) + \tfrac{15}{16}\tilde{x}^4\tilde{y}^2(231x^4y^2 - 21x^4r^2 - 126x^2y^2r^2 + 14x^2r^4 + 7y^2r^4 - r^6) \\ &+ \tfrac{15}{16}\tilde{x}^4\tilde{z}^2(231x^4z^2 - 21x^4r^2 - 126x^2z^2r^2 + 14x^2r^4 + 7z^2r^4 - r^6) + \tfrac{15}{8}\tilde{x}^2\tilde{y}^2\tilde{z}^2(231x^2y^2z^2 - 63x^2y^2r^2 + 2r^6). \end{aligned} \tag{B27}$$

This leads to the first nonleading contribution to the non-Keplerian part of the potential of a solid cube:

$$\Phi_6 = \frac{1}{1344}\frac{Ma^6}{r^{13}}\left[30r^6 - 105r^2\left(x^4 + y^4 + z^4\right) + 77\left(x^6 + y^6 + z^6\right)\right]. \tag{B28}$$

As before, we can check the expression by computing the spherical average, and indeed $\int\int \Phi_6 \sin\theta\, d\phi\, d\theta = 0$.

The x-component of the force derived from the potential as given above follows from $F_x = (\delta/\delta x)\Phi$ as

$$F_x(\boldsymbol{r}) = \frac{M}{r^3}x - \frac{7}{96}\frac{Ma^4}{r^{11}}x\left[5x^2r^2 - 6x^4 - 3\left(y^4 + z^4\right)\right] - \frac{1}{192}\frac{Ma^6}{r^{13}}x\left[-23x^2r^2 + 25x^4 + 8\left(y^4 + z^4\right)\right]. \tag{B29}$$

The y- and z-components of the force follow by cyclic permutation of $\{x, y, z\}$.

APPENDIX C

A LINEAR DENSITY GRADIENT

Consider a box with sides of length a and with density

$$\rho'(\boldsymbol{r}') = \rho'(x', y', z') = \begin{cases} \rho'_0 + \dfrac{1}{a}(\boldsymbol{r}'\cdot\hat{\boldsymbol{n}})\rho'_1, & |x'| < \frac{1}{2}a,\ |y'| < \frac{1}{2}a, \text{ and } |z'| < \frac{1}{2}a, \\ 0, & \text{elsewhere,} \end{cases} \tag{C1}$$

where ρ'_0 and ρ'_1 are constants with physical dimension of density and

$$\hat{\boldsymbol{n}} = (n_x, n_y, n_z) = (\sin\chi\cos\varphi, \sin\chi\sin\varphi, \cos\chi) \tag{C2}$$

is a unit vector pointing along the density gradient. The center of mass of the box is at position $\boldsymbol{R}' = (X', Y', Z')$, where

$$\boldsymbol{R}' = \frac{1}{M}\int\int\int d\boldsymbol{r}'\rho'(\boldsymbol{r}')\boldsymbol{r}' = \frac{a}{12}\left(\frac{\rho'_1}{\rho'_0}\right)\hat{\boldsymbol{n}} \tag{C3}$$

and the total mass in the box

$$M = \int\int\int d\boldsymbol{r}'\rho'(\boldsymbol{r}') = a^3\rho'_0. \tag{C4}$$

Transforming to a coordinate system $(\tilde{x}, \tilde{y}, \tilde{z})$ centered on the center of mass and having coordinate axes parallel to the (x', y', z') axes gives the density in the new coordinate system as

$$\rho(\tilde{x}, \tilde{y}, \tilde{z}) = \begin{cases} \tilde{\rho}_0 + (1/a)(\tilde{\boldsymbol{r}}\cdot\hat{\boldsymbol{n}})\tilde{\rho}_1, & \tilde{\boldsymbol{r}} \in \tilde{V}, \\ 0, & \text{elsewhere,} \end{cases} \tag{C5}$$

where the homogeneous part of the density $\tilde{\rho}_0$ is given by

$$\tilde{\rho}_0 = \rho'_0 + \frac{1}{a}(\boldsymbol{R}'\cdot\hat{\boldsymbol{n}})\rho'_1 = \rho'_0\left[1 + \frac{1}{12}\left(\frac{\rho'_1}{\rho'_0}\right)^2\right], \tag{C6}$$

the coefficient ρ_1 for the linearly changing part by

$$\tilde{\rho}_1 = \rho'_1, \tag{C7}$$

and the volume $\tilde{V}$ of the box is described in the new coordinates by

$$\tilde{x}_- < \tilde{x} < \tilde{x}_+, \qquad \tilde{y}_- < \tilde{y} < \tilde{y}_+, \qquad \text{and } \tilde{z}_- < \tilde{z} < \tilde{z}_+ \tag{C8}$$

with

$$\begin{aligned} \tilde{x}_\pm &= \pm\tfrac{1}{2}a - (a/12)(\rho'_1/\rho'_0)\sin\chi\cos\varphi, \\ \tilde{y}_\pm &= \pm\tfrac{1}{2}a - (a/12)(\rho'_1/\rho'_0)\sin\chi\sin\varphi, \\ \tilde{z}_\pm &= \pm\tfrac{1}{2}a - (a/12) - (\rho'_1/\rho'_0)\cos\chi. \end{aligned} \tag{C9}$$

We now develop the potential in a series as given by equation (B8):

$$\Phi(\boldsymbol{r}) = \sum_{i=0}^{\infty}\Phi_i(\boldsymbol{r}), \tag{C10}$$

with

$$\Phi_i(\boldsymbol{r}) = \frac{1}{r^{2i+1}}\int_{\tilde{x}_-}^{\tilde{x}_+}d\tilde{x}\int_{\tilde{y}_-}^{\tilde{y}_+}d\tilde{y}\int_{\tilde{z}_-}^{\tilde{z}_+}d\tilde{z}\left[\tilde{\rho}_0 + \frac{1}{a}(\tilde{\boldsymbol{r}}\cdot\hat{\boldsymbol{n}})\tilde{\rho}_1\right]Q_i(\boldsymbol{r}, \tilde{\boldsymbol{r}}) \tag{C11}$$

The monopole contribution gives

$$\Phi_0(\boldsymbol{r}) = \frac{1}{r}\left\{\tilde{\rho}_0 a^3 + \frac{\tilde{\rho}_1}{2a}\left[\sin\chi\cos\varphi\left(\tilde{x}_+^2 - \tilde{x}_-^2\right) + \sin\chi\sin\varphi\left(\tilde{y}_+^2 - \tilde{y}_-^2\right) + \cos\chi\left(\tilde{z}_+^2 - \tilde{z}_-^2\right)\right] a^2\right\}$$

$$= \frac{1}{r}\left[\tilde{\rho}_0 a^3 - \left(\frac{\tilde{\rho}_1}{2a}\right)\left(\frac{a^4}{6}\right)\left(\frac{\rho_1'}{\rho_0'}\right)\right] = \frac{1}{r}\rho_0' a^3 = \frac{M}{r} \tag{C12}$$

as expected, since the monopole contribution is independent of the coordinate system used (see Jackson 1975). The dipole contribution can be similarly computed and results in

$$\Phi_1 = \frac{1}{r^3}\int d\tilde{\boldsymbol{r}}\left[\tilde{\rho}_0 + \frac{1}{a}(\tilde{\boldsymbol{r}}\cdot\hat{\boldsymbol{n}})\tilde{\rho}_1\right](\boldsymbol{r}\cdot\tilde{\boldsymbol{r}}) = 0, \tag{C13}$$

again as expected, since the gravitational dipole moment vanishes for a density distribution if the expansion is performed around the center of mass of the system (see Goldstein 1980).

Evaluating integrals such as those in equation (C13) becomes progressively more tedious. A useful result from equation (C9) is

$$\tilde{x}_+^2 - \tilde{x}_-^2 = -\frac{a^2}{6}\frac{\rho_1'}{\rho_0'}\sin\chi\cos\varphi,$$

$$\tilde{y}_+^2 - \tilde{y}_-^2 = -\frac{a^2}{6}\frac{\rho_1'}{\rho_0'}\sin\chi\sin\varphi, \tag{C14}$$

$$\tilde{z}_+^2 - \tilde{z}_-^2 = -\frac{a^2}{6}\frac{\rho_1'}{\rho_0'}\cos\chi.$$

Similarly

$$\tilde{x}_+^3 - \tilde{x}_-^3 = \frac{a^3}{4}\left[1 + \frac{1}{12}\left(\frac{\rho_1'}{\rho_0'}\right)^2 n_x^2\right], \tag{C15}$$

with $n_x = \sin\chi\cos\varphi$, and the $\tilde{y}$ and $\tilde{z}$ generalizations following with $n_y = \sin\chi\sin\varphi$ and $n_z = \cos\chi$. A few higher powers give

$$\tilde{x}_+^4 - \tilde{x}_-^4 = -\frac{a^4}{12}\frac{\rho_1'}{\rho_0'}n_x\left[1 + \frac{1}{36}\left(\frac{\rho_1'}{\rho_0'}\right)^2 n_x^2\right] \tag{C16}$$

and

$$\tilde{x}_+^5 - \tilde{x}_-^5 = \frac{a^5}{16}\left[1 + \frac{5}{18}\left(\frac{\rho_1'}{\rho_0'}\right)^2 n_x^2 + \frac{5}{1296}\left(\frac{\rho_1'}{\rho_0'}\right)^4 n_x^4\right]; \tag{C17}$$

$$\Phi_2 = \frac{1}{r^5}\int d\tilde{\boldsymbol{r}}\left[\tilde{\rho}_0 + \frac{1}{a}(\tilde{\boldsymbol{r}}\cdot\boldsymbol{n})\tilde{\rho}_1\right]\left[\frac{3}{2}(\boldsymbol{r}\cdot\tilde{\boldsymbol{r}})^2 - \frac{1}{2}r^2\tilde{r}^2\right]$$

$$= \frac{Ma^2}{288r^5}\left(\frac{\rho_1'}{\rho_0'}\right)^2\left[r^2 - 3(\boldsymbol{r}\cdot\hat{\boldsymbol{n}})^2\right] \tag{C18}$$

Again, we have an explicit check on this equation from the fact that the spherical average vanishes: $\int\int\Phi_2\sin\theta\,d\phi\,d\theta = 0$.

The octopole contribution to the potential is found to be

$$\Phi_3 = \frac{Ma^3}{8640r^7}\left\{36\left[3r^2(\boldsymbol{R}_1\cdot\boldsymbol{N}_1) - 5(\boldsymbol{R}_3\cdot\boldsymbol{N}_1)\right]\left(\frac{\rho_1'}{\rho_0'}\right)\right.$$

$$\left. + 5\left[-3r^2(\boldsymbol{R}_1\cdot\boldsymbol{N}_1) + 15(\boldsymbol{R}_2\cdot\boldsymbol{N}_2)(\boldsymbol{R}_1\cdot\boldsymbol{N}_1) - 10(\boldsymbol{R}_3\cdot\boldsymbol{N}_3)\right]\left(\frac{\rho_1'}{\rho_0'}\right)^3\right\}, \tag{C19}$$

where we have introduced the concise notation

$$\boldsymbol{R}_k = (x^k, y^k, z^k) \tag{C20}$$

and

$$\boldsymbol{N}_k = (n_x^k, n_y^k, n_z^k). \tag{C21}$$

In particular, $\boldsymbol{R}_1 = \boldsymbol{r}$ and $\boldsymbol{N}_1 = \boldsymbol{n}$, with $(\boldsymbol{R}_1 \cdot \boldsymbol{R}_1) = r^2$ and $(\boldsymbol{N}_1 \cdot \boldsymbol{N}_1) = 1$.

Finally, the hexadecapole contribution to the potential is given by

$$\begin{aligned}\Phi_4 = \frac{Ma^4}{276480 r^9}\Bigg\{ & 2016\left[3r^4 - 5(\boldsymbol{R}_2 \cdot \boldsymbol{R}_2)\right] + 96\left[-12r^4 + 15(\boldsymbol{R}_2 \cdot \boldsymbol{R}_2) + 5(\boldsymbol{R}_1 \cdot \boldsymbol{N}_1)(\boldsymbol{R}_3 \cdot \boldsymbol{N}_1) + 15(\boldsymbol{R}_2 \cdot \boldsymbol{P}_2)\right]\left(\frac{\rho_1'}{\rho_0'}\right)^2 \\ & + 5\left[9r^4 + 12r^4(\boldsymbol{N}_2 \cdot \boldsymbol{N}_2) - 48r^2(\boldsymbol{R}_2 \cdot \boldsymbol{N}_2) + 30r^2(\boldsymbol{R}_1 \cdot \boldsymbol{N}_1)^2 + 48r^2(\boldsymbol{R}_2 \cdot \boldsymbol{N}_4) - 60(\boldsymbol{R}_1 \cdot \boldsymbol{N}_1)^4 + 150(\boldsymbol{R}_2 \cdot \boldsymbol{N}_2)(\boldsymbol{R}_1 \cdot \boldsymbol{N}_1)^2\right. \\ & - 51(\boldsymbol{R}_2 \cdot \boldsymbol{N}_2)^2 + 12(\boldsymbol{R}_2 \cdot \boldsymbol{R}_2)(\boldsymbol{N}_2 \cdot \boldsymbol{N}_2) - 200(\boldsymbol{R}_3 \cdot \boldsymbol{N}_3)(\boldsymbol{R}_1 \cdot \boldsymbol{N}_1) \\ & \left. + 48r^2(\boldsymbol{R}_2 \cdot \boldsymbol{N}_4) + 48(\boldsymbol{R}_4 \cdot \boldsymbol{N}_2) + 86(\boldsymbol{R}_4 \cdot \boldsymbol{N}_4)\right]\left(\frac{\rho_1'}{\rho_0'}\right)^4\Bigg\},\end{aligned} \tag{C22}$$

where

$$\boldsymbol{P} = \boldsymbol{R} \times \boldsymbol{N} \tag{C23}$$

and $\boldsymbol{P}_2$ is defined in analogy with $\boldsymbol{R}_2$ and $\boldsymbol{N}_2$, i.e.,

$$\boldsymbol{P}_2 = \left[(yn_z - zn_y)^2, (zn_x - xn_z)^2, (xn_y - yn_x)^2\right]. \tag{C24}$$

APPENDIX D
MEASUREMENT OF FORCE-CALCULATION ERRORS

I. METHODS

We test our force-calculation algorithm by comparing it with a simple direct-sum algorithm. As input for a force calculation test, we need:

1. Coordinates $\boldsymbol{R}_i$ for a set of N massive particles, which generate the field.
2. Coordinates $\boldsymbol{r}_i$ for a set of n sample positions, at which the field is evaluated.
3. A softening parameter ε to be used by both force calculation algorithms.
4. A tolerance parameter θ for the tree algorithm.

At each sample position $\boldsymbol{r} = \boldsymbol{r}_i$, we obtain approximate values $\Phi_t(\boldsymbol{r})$, $\boldsymbol{a}_t(\boldsymbol{r})$ from the tree algorithm, and exact values $\Phi(\boldsymbol{r})$, $\boldsymbol{a}(\boldsymbol{r})$ by direct sum. From these we can then evaluate such quantities as the error in the potential,

$$\Delta\Phi \equiv \Phi_t - \Phi, \tag{D1}$$

and the error in the acceleration,

$$\Delta a \equiv |\Delta \boldsymbol{a}|, \qquad \Delta \boldsymbol{a} \equiv \boldsymbol{a}_t - \boldsymbol{a}. \tag{D2}$$

II. TESTS

Four different mass distributions were studied in detail:

1. A uniform-density "ball" of mass $M \equiv 1$ and radius $R \equiv 1$.
2. A King model (King 1966) with mass $M \equiv 1$, core radius $R_c \simeq 0.103$, tidal radius $R_t \simeq 3.47$, and dimensionless central potential $W_c \equiv 7$.
3. A disk with exponential surface density profile, mass $M \equiv 1$, inverse scale $\alpha = 4.0$, and vertical thickness $\langle z^2 \rangle^{1/2} = 0.025$.
4. A Soneira and Peebles (1977) clustering hierarchy, with radius $R = 1$ overall, $L = 8$ levels, $\eta = 4$ branches per level, and a power-law correlation function slope of $\gamma = 1.8$, natch.

In the first three cases, coordinates for $N = 4096$ massive points and $n = 512$ test positions were chosen by picking points at random from the spatial density $\rho(\boldsymbol{r})$, interpreted as a one-point probability distribution function. Since different particles are

chosen independently, this procedure builds Poissonian fluctuations into the mass distribution. In the fourth case, the complete hierarchy of $\eta^{L-1} = 16{,}384$ points was sampled to obtain disjoint subsets of $N = 4096$ massive points and $n = 512$ test positions.

For each mass distribution, four tolerance parameters were tested: $\theta = 1, 0.7, 0.5, 0.3$. Each test was run both with and without the quadrupole correction. Most tests were run with softening parameter $\varepsilon = 0$, but other values were also tried.

REFERENCES

Aarseth, S. J. 1985, in *Multiple Time Scales*, ed. Brackill and Cohen (New York: Academic), p. 377.
Ahmad, A., and Cohen, L. 1973, *J. Comput. Phys.*, **12**, 389.
Appel, A. 1981, *B.A. thesis*, Princeton University.
_____. 1985, *SIAM J. Sci. Stat. Comput.*, **6**, 85.
Barnes, J. 1986, in *The Use of Supercomputers in Stellar Dynamics*, ed. P. Hut and S. McMillan (Berlin: Springer-Verlag), p. 175.
_____. 1989, *J. Comput. Phys.*, in press.
Barnes, J., and Hut, P. 1986, *Nature*, **324**, 446.
Benz, W., Bowers, R. L., Cameron, A. G. W., and Press, W. H. 1989, *Ap. J.*, in press.
Carrier, J., Greengard, L., and Rokhlin, V. 1988, *SIAM J. Sci. Stat. Comput.*, **9**, 669.
Chandrasekhar, S. 1943, *Rev. Mod. Phys.*, **15**, 1.
Dejonghe, H., and Hut, P. 1986, in *The Use of Supercomputers in Stellar Dynamics*, ed. P. Hut and S. McMillan (Berlin: Springer-Verlag), p. 212.
Dongarra, J. 1988, *Argonne National Lab. Math. Comput. Sci. Division, Tech. Memo*, No. 23.
Efstathiou, G., Davis, M., Frenk, C. S., and White, S. D. M. 1985, *Ap. J. Suppl.*, **57**, 241.
Goldstein, H. 1980, *Classical Mechanics* (Reading: Addison-Wesley).
Greengard, L., and Rokhlin, V. 1987, *J. Comput. Phys.*, **73**, 325.
Heggie, D., Goodman, J., and Hut, P. 1989, in preparation.
Hernquist, L. 1987, *Ap. J. Suppl.*, **64**, 715.
_____. 1989, *J. Comput. Phys.*, in press.
Hernquist, L., and Barnes, J. 1989, *Ap. J.*, submitted.
Hernquist, L., and Katz, N. 1989, *Ap. J. Suppl.*, in press.
Hillis, D. 1985, *The Connection Machine* (Cambridge: MIT).
Hut, P., Makino, J., and McMillan, S. 1988, *Nature*, **336**, 31.
Hockney, R. W., and Eastwood, J. W. 1981, *Computer Simulation Using Particles* (New York: McGraw-Hill).
Holmberg, E. 1941, *Ap. J.*, **94**, 385.
Jackson, J. D. 1975, *Classical Electrodynamics* (New York: John Wiley).
Jernigan, J. G. 1985, in *I.A.U. Symposium 113, Dynamics of Star Clusters*, ed. J. Goodman and P. Hut (Dordrecht: Reidel), p. 275.
Katzenelson, J. 1989, *SIAM J. Stat. Sci. Comput.*, in press.
Kellogg, O. D. 1929, *Foundations of Potential Theory* (Berlin: Springer-Verlag).
King, I. 1966, *A.J.* **76**, 64.
Lecar, M. 1968, *Bull. Astr.*, **3**, 91.
Makino, J. 1989, *J. Comput. Phys.*, in press.
Miller, R. H. 1964, *Ap. J.*, **140**, 250.
_____. 1978, *Ap. J.*, **223**, 1922.
Porter, D. 1985, *Ph.D. thesis*, University of California, Berkeley.
Press, W. H. 1986, in *The Use of Supercomputers in Stellar Dynamics*, ed. P. Hut and S. McMillan (Berlin: Springer-Verlag), p. 184.
Soneira, R. M., and Peebles, P. J. E. 1977, *Ap. J.*, **211**, 1.
van Albada, T. S. 1982, *M.N.R.A.S.*, **201**, 939.
van Albada, T. S., and van Gorkom, J. 1977, *Astr. Ap.*, **54**, 121.
Villumsen, J. V. 1982, *M.N.R.A.S.*, **199**, 493.

JOSHUA BARNES and PIET HUT: Institute for Advanced Study, Princeton, NJ 08540