OLD DOMINION UNIVERSITY RESEARCH FOUNDATION

DEPARTMENT OF PHYSICS AND GEOPHYSICAL SCIENCES
SCHOOL OF SCIENCES AND HEALTH PROFESSIONS
OLD DOMINION UNIVERSITY
NORFOLK, VIRGINIA

Technical Report PGSTR-PH77-66

COMPUTER SIMULATION OF PLASMA AND N-BODY PROBLEMS

(NASA-CR-155517) COMPUTER SIMULATION OF PLASMA AND N-BODY PROBLEMS Final Report, 1 Jun. 1974 - 31 Dec. 1977 (Old Dominion Univ. Research Foundation) 64 p HC A04/MF A01 CSCL 03B G3/90
N78-15972
Unclas
57753

By

Wynford L. Harries

John B. Miller

and

Christopher Costner

Final Report
For the period June 1, 1974 - December 31, 1977

Prepared for the
National Aeronautics and Space Administration
Langley Research Center
Hampton, Virginia

Under
Research Grant NSG 1040
Dr. Frank Hohl, Technical Monitor
Space Systems Division

December 1977

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

(cont'd.)

LIST OF FIGURES - CONCLUDED

COMPUTER SIMULATION OF PLASMA AND N-BODY PROBLEMS

By

Wynford L. Harries[1], John B. Miller[2], and Christopher Costner[3]

INTRODUCTION

In recent years, large N-body computer simulations (Miller and Prendergast, 1968; Hohl and Hockney, 1969) have become an important tool in investigating the structure of spiral galaxies, especially in determining the development of large-scale instabilities resulting in spiral and bar formation. Until recently, most of these simulations used essentially two-dimensional models with the "stars" confined to the plane of the galactic disk (Miller, Prendergast, and Quirk, 1970; Hohl, 1971). These simulations have shown that the disks of stars have a tendency for the development of fast growing nonaxisymmetric instabilities resulting in bar formation. The bar instabilities occur even for velocity dispersions that are considerably larger than those found in the solar neighborhood or those predicted by Toomre (1964) as being locally stabilizing. Because of the difficulties in solving the highly nonlinear problem, global instability studies of disks of stars have been primarily numerical. Some limited work has been done for uniformly rotating disks (Hunter, 1963; Kalnajs, 1972), but generally linear stability analyses were used in the studies of disks of stars.

Any spiral structure in computer-generated galaxies is generally short lived and the final state is a rotating bar. The bar thus obtained rotates more slowly than the stars. For one case investigated by Hohl (1971), the bar rotates at 2.25τ and the stars rotate at 1.5τ where τ is the rotational period of the initial disk.

---

[1] Professor, Department of Physics and Geophysical Sciences, Old Dominion University, Norfolk, Virginia 23508.

[2] Research Associate, Old Dominion University Research Foundation, Norfolk, Virginia 23508.

[3] Graduate Research Assistant, Department of Physics and Geophysical Sciences, Old Dominion University, Norfolk, Virginia 23508.

It has been argued that core/halo components have a stabilizing effect on galaxies and result in longer lived spiral structure (Ostriker and Peebles, 1973). However, numerical experiments with large fixed stellar components representing the core/halo component (Hohl, 1970; Hockney and Brownrigg, 1974) show that multiarmed spiral structure develops and persists for many rotations but only in an evolving manner. That is, the spiral structure is either wound up into a tight pattern or it is wound up and then reappears again. A recent study of the effect of fixed core/halo components (Hohl, 1976) does show that the bar instability is indeed inhibited by a sufficiently large fixed component.

The purpose of the present study is twofold. First, we want to determine the effect of a self-consistent (rather than fixed) core/halo component. This will show whether there are any instabilities (such as "two-stream") or other important interactions present that may be suppressed with a fixed core. Second, we want to determine the effects of finite thickness of the disk and of three-dimensional essentially spherical core/halo components.

MODEL

The model used for the present galaxy simulations consists of 100,000 representative stars that move inside an array of cells. For the disk simulations the stars are confined to move in the plane of the disk represented by a 64 × 64 active array. In the three-dimensional simulation the stars move inside a 64 × 64 × 16 array of cells. The sum of the stars inside each cell defines the mass density at the center of each cell. Fast Fourier transform methods are used to obtain the gravitational field at the center of each cell for a given density distribution. The force acting on a particular star is determined by bilinear (or trilinear) interpolation from the values of the gravitational fields at the surrounding 4 (or 8) cell centers. After the force acting on a star is determined, it is advanced by a small timestep, the new density is recalculated and the process is continued until the desired evolution is achieved. If a star leaves the array of cells, approximate methods are used to determine the force acting on the star. Details of the disk model are described in detail by Hohl and Hockney (1969) and by Hohl (1970). The extension of the model to three dimensions is described in the appendix.

RESULTS

Observational evidence (deVoucouleurs 1959; Freeman 1970; Kormendy 1977) indicates that the luminosity (and presumably the density) in the outer regions of many spiral and SO galaxies decreases exponentially with radius. Also, previous simulations (Hohl 1971) showed that intially unstable stellar disks evolved into stable systems with radial density variations that closely approximated the sum of two exponentials. The inner exponential with a scale length of about 1 kpc describes the non- or slowly-rotating spheroidal or core component and the remaining exponential with a scale length of about 8 kpc describes the extended disk population. Thus, it seems reasonable to use an exponential density variation for the disk of the present computer simulations. Similarly, the central core used is described by an exponential density variation.

Figure 1 illustrates the evolution of a disk of 100,000 stars with an initially exponential surface density distribution $\mu(r) = \mu_o e^{-r/2}$ with a cutoff at $r = 10$ kpc. The initial angular velocity of the disk was obtained from

$$\omega^2 = \omega_o^2 + \frac{1}{r\mu(r)}\frac{\partial}{\partial r}[\mu(r)\ \sigma_r^2(r)] + \frac{1}{r^2}[\sigma_r^2(r) - \sigma_\theta^2(r)] \tag{1}$$

with

$$\sigma_\theta(r) = \frac{\kappa(r)}{2\omega_o(r)}\sigma_r(r) \tag{2}$$

Here, $\omega_o(r)$ is the angular velocity required to balance the cold (zero velocity dispersion) disk, $\omega(r)$ is the actual angular velocity, and $\kappa(r)$ is the epicyclic frequency. The initial value of the radial velocity dispersion σ_r was taken to be that determined by Toomre (1964) as the minimum required to stabilize all axisymmetric instabilities,

$$\sigma_r(r) = \sigma_{r,min} = 3.36\ G\mu(r)/\kappa(r) \tag{3}$$

The time t is given in rotational periods $\left(\frac{2\pi}{\omega_o}\right)$ of the cold disk at a radius of 5 kpc, that is half way to the edge of the initial disk.

As expected (Hohl 1970, 1971), only the small-scale instabilities are prevented by $\sigma_r = \sigma_{r,min}$ and the system quickly forms a two-arm spiral which eventually tends to evolve into a rotating bar. The evolution of the azimuthally averaged radial density variation for this system is shown in Fig. 2. As previously observed (Hohl 1970, 1971) the eventual density variation approaches one which can be closely approximated by the sum of two exponentials. One exponential describing the central core component and the other describing the extended disk. The evolution of the radial velocity is such that there is some heating near the center, and a considerable increase in the velocity dispersion for stars expanding into the extended disk component. Numerous other diagnostics have been performed on the system. For example, Fig. 3 shows the time evolution of the moment of inertia I divided by the moment of inertia at $t = 0$, and a similar ratio for the angular momentum. P. As can be seen, P is conserved in the simulation but I is still increasing at a near linear rate after three rotations. The evolution of various components of the total kinetic energy divided by the total potential energy is shown in Fig. 4. The components T_r and T_θ represent the kinetic energies due to the velocity dispersions $\sum_i m_i \sigma_{i_\theta}^2$ and $\sum_i m_i \sigma_{i_r}^2$, respectively, while T_{cir} is the kinetic energy of rotation. Note that the ratio of the kinetic energy in rotation to the absolute value of the total gravitational energy of the system is approaching the value 0.14 predicted by Ostriker and Peebles (1973) for stability. At the same time, there occurs considerable heating of the system.

One of the aims of the present study is to determine the effect of adding the third degree of freedom by allowing a finite thickness of the exponential disk. Using again an exponential projected surface density variation $\mu = \mu_o e^{-r/2}$ the stars are now distributed in the z-direction according to one-dimensional distribution $\mathrm{sech}^2 z/c$ where c is a parameter determined from $\mu(r)$, (Hohl 1967). The central thickness of the disk is 2 kpc and the density is cut off at z_1 given by

$$\sqrt{1 - \left(\frac{r}{R}\right)^2}\ \mathrm{sech}^2\left(\frac{z_1}{c}\right) = 0.1 \tag{4}$$

where $R = 10$ kpc is the radius of the disk. The radial and azimuthal velocity components are determined in a manner similar to that for the infinitesimally thin disk and the z-component of the velocity dispersion is determined by a force balance in the z-direction. Note also that all initial velocities are truncated such that stars have kinetic energies no greater than that which would allow them to reach the boundary of the system in the gravitational potential at $t = 0$.

Figure 5 shows a side view of the initial disk and the evolution for up to 3 rotations. Note the rapid expansion in the plane of the disk. This is the result of the bar instability as shown in Fig. 6 which gives the evolution of the disk projected in the x-y plane. Note that the evolution is very similar to that shown in Fig. 1 for the infinitely thin disk. Similarly the evolution of the surface density variation and the increases in the moment of inertia are nearly identical to those shown in Figs. 2 and 3 for the thin disk. The ratio of the various kinetic energy components for the total potential energy are shown in Fig. 7 for the finite thickness disk. Note that again the evolution is similar to that for the infinitely thin disk as shown in Fig. 4. An additional variable, the z-component of the kinetic energy, is given in Fig. 7 and shows that since this component remains small compared to the others one would expect little difference in the evolution of the finite thickness disk when compared to the infinitely thin disk.

As shown in Figs. 1 and 6, exponential disks with velocity dispersion ($Q \approx 1$) are violently unstable to the bar-forming instability. Previous work (Hohl, 1976) with a superimposed fixed (nonself-consistent) central mass distribution indicated a stabilizing effect toward the bar-forming instability. A more realistic simulation is to allow core-disk interaction, thus, presently we are interested in the stabilizing effects of a completely self-consistent core or "spheroid" component. Again, the effect is investigated for both the infinitesimally thin disk (two-dimensional) and for the three-dimensional disk.

For the core-disk system, 50 percent of the mass (50,000 stars) is contained in the nonrotating core and the remaining mass (50,000 stars) is contained in the disk. The disk component is again given the surface density variation $\mu_{disk} = \mu_o e^{-r/2}$ whereas, the initial nonrotating core component is given a density variation $\mu_{core} = \mu_o^{-r/0.5}$. Note that the disk and core density are cut off at $r = 10$ kpc and $r = 3.5$ kpc, respectively. The initial velocity dispersion and rotation of the disk is obtained by again using Eq. (1), (2), and (3) with $\mu = \mu_{disk}$. Similarly, as before, the z-dimensions of the disk are determined from Eq. (4). The initial velocity dispersion of the nonrotating core was obtained by taking $\sigma_\theta = \sigma_r$ and simply balancing the core in the presence of the disk. In order to assure that the core component was in a stable state at the start of the core-disk simulation, the core was allowed to evolve for several rotational periods $(2\pi/\omega_o$ at 5 kpc) with the disk component held fixed. Starting from these initial conditions, the system evolved as shown in Fig. 8. Note that even though a two-arm spiral structure still forms, the system as a whole evolves in a much less violet manner than that displayed in Fig. 1. This can also be seen in Fig. 9 which shows the evolution of the surface mass density for both the core and disk component. Note that with the exception of a slow outward diffusion of stars near the edge, the core remains essentially stationary, while the disk component displays the outward shift of mass generally associated with bar formation. Similar information is contained in Fig. 10 which displays the evolution of the radial velocity dispersion for the core and disk component. Note the sharp increase in the velocity dispersion at $r = 2$ kpc which is associated with a marked reduction in the angular momentum of the disk in this region.

In general, the simulations show that the formation of bars or two-armed spirals results in moving angular momentum outward to larger radii. Fig. 11 shows a marked reduction in the rate of increase of the moment of inertia when compared to the disks without a central core component.

The final system investigated is that of a three-dimensional exponential disk with a three-dimensional core or spheroid component. The spatial distribution of the stars for the disk component is obtained, as was done for the disk shown in Fig. 5 and 6, except that now the disk contains only 50,000 stars. For the nonrotating central core the density is given by $\rho = \rho_o e^{-\zeta/0.5}$ where $\zeta = x^2 + y^2 + (z/c)^2$ with $c = 5/7$. The density is cut off at $\zeta = 7$. Thus, the central core or spheroid has an axis ratio of 7:5. Again the Gaussian velocity dispersion for the core is obtained by a simple balance of the self-gravity of the total system. The velocity dispersion for the disk component is generated, as was done for the system shown in Fig. 6. Before initiating the simulation of the combined core-disk system, the core was allowed to evolve for several rotations (free-fall periods) to assure that no instabilities or other problems associated with the core component were present.

Figure 12 shows the evolution of the system perpendicular to the equatorial plane. Note the remarkable stability of the system when compared to the disk without the central core in Fig. 5. The evolution of the system in the equatorial plane is shown in Fig. 13 and displays the development of a comparatively weak two-arm spiral structure. It should be noted that because of the allowed initial relaxation, the core components of the two core-disk systems investigated here are expected to closely satisfy the collisionless Boltzmann equation. The same is not necessarily true for the disk component since satisfying equation (1) only assures a balance of forces at $t = 0$. Also, we know that for a stellar disk $\sigma_r = \sigma_{r,min}$ does not assure stabilization of global nonaxisymmetric instabilities (Toomre, 1974; 1977). However, since one would hardly expect nature to generate a galaxy initially in an exact stable stationary state, and since we are interested in the further development of instabilities and the final state toward which the system evolves, an exact stationary and stable initial state is not necessary.

The evolution of the azimuthally averaged projected surface mass density for the three-dimensional core-disk system is shown in Fig. 14 and is nearly identical to that of the two-dimensional core-disk system shown in Fig. 9. Note that there is very little change in the density for the core with the exception of a slight outward diffusion near the edge. Azimuthally averaged values of the total density variation in the z-direction are shown in Fig. 15 for various values of r. Some of the fluctuations shown may be due to the relatively small sampling volume used. If we look at the evolution of the radial velocity dispersion shown in Fig. 16 we see that (as expected) the velocity dispersion for the two-dimensional core (Fig. 10) is higher. Also, the large increase in the velocity dispersion of the disk near $r = 2$ kpc does not occur for the three-dimensional disk. Associated with this is the fact that there is very little change in the radial angular moment distribution during the evolution of the 3-D core-disk system, whereas, considerable outward shift of angular momentum occurs for the 2-D core-disk system. These results indicate that the global bar instability is much weaker for the 3-D system as for the 2-D system, as can be seen by comparing Figs. 8 and 13.

The time evolution of the various kinetic energy ratios for the 3-D disk-core system is shown in Fig. 17. As can be seen, there is little change in the value of the various components during the evolution. Note that the value of the ratio of the kinetic energy in rotation to the total potential energy of the system is slightly higher than the value of the 0.14 predicted for stability by Ostriker and Peebles (1973). Also, the moment of inertia increases by only about one third of that shown in Fig. 11 for the 2-D system. As was the case for all four systems investigated, the angular momentum was conserved to a sufficient degree of accuracy.

REFERENCES

Freeman, K. C. 1970, Ap. J., 160, 811.

Hockney, R. W., and Brownrigg, D. R. K. 1974, M.N.R.A.S., 167, 351.

Hohl, F. 1967, Ap. J., 147, 1164.

Hohl, F. 1970, NASA TR R-343, Clearinghouse for Federal Scientific Information, Springfield, Virginia 22151.

Hohl, F. 1971, Ap. J., 168, 343.

Hohl, F. 1976, Astron. J., 81, 30.

Hunter, C. 1963, M.N.R.A.S., 126, 299.

Kalnajs, A. 1972, Ap. J., 175, 63.

Kormendy, J. 1977, Ap. J., 217, 406.

Ostriker, J. P., and Peebles, P. J. E., 1973, Ap. J., 186, 467.

Toomre, A. 1964, Ap. J., 139, 1217.

Toomre, A. 1974, in IAU Highlights of Astronomy, vol. 3, ed. G. Coutopoulos (Boston: Reidel), p. 455.

Toomre, A. 1977, Ann. Rev. Astron. Astrophys., 15, 437.

Voucouleurs, G. de 1954, Hdb. d. Phys., 53 311.

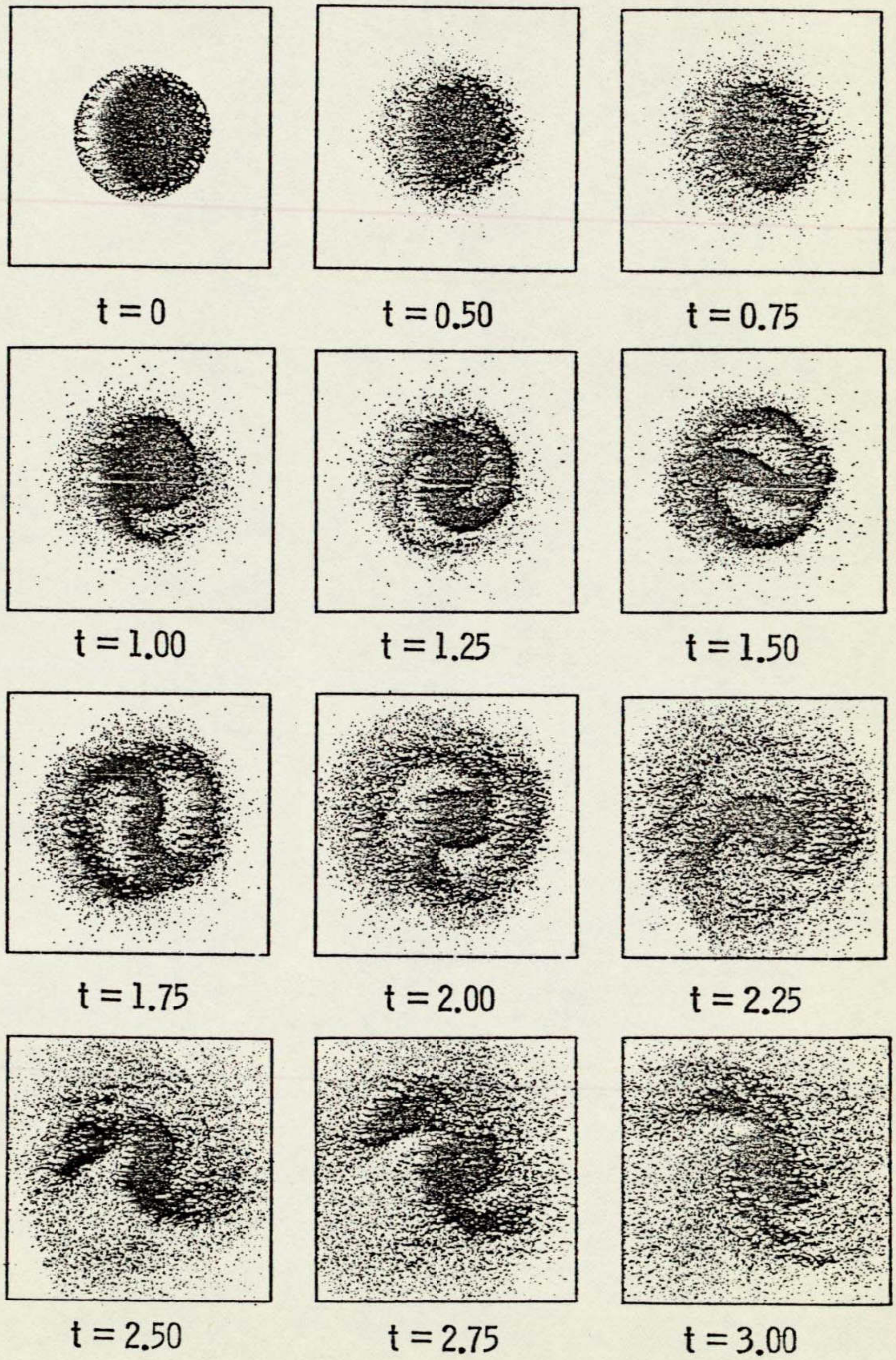

Figure 1.- Evolution of an initially balanced, infinitesimally thin disk of 100,000 stars with an exponential radial density variation. The stars have an initial density variation given by Toomre's criterion.

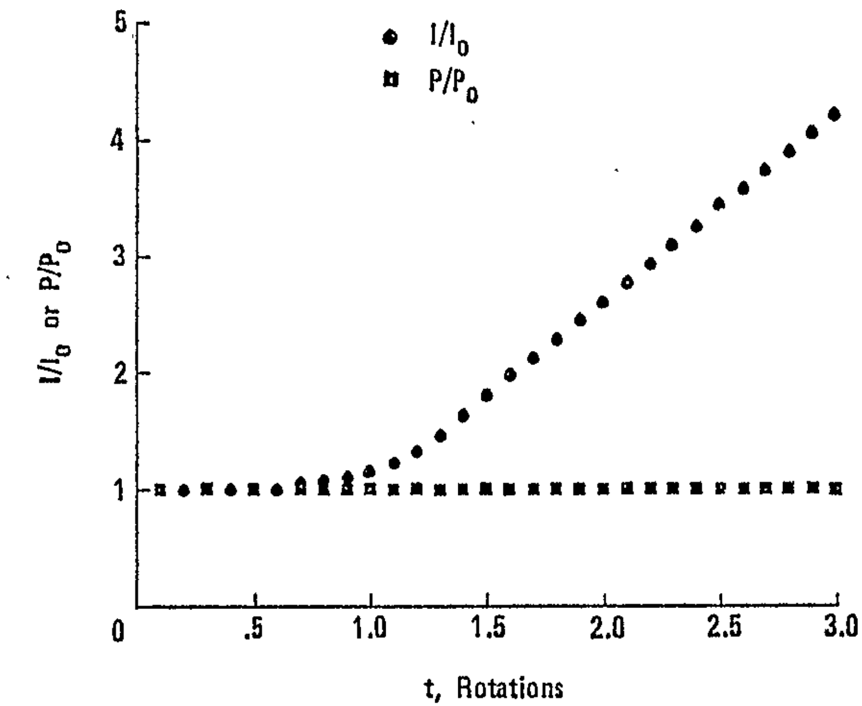

Figure 3.- Time evolution of the angular momentum (P) and the moment of inertia (I) for the unstable disk shown in figure 1. Note the rapid increase in I as the bar begins to form at t ≈ 1.

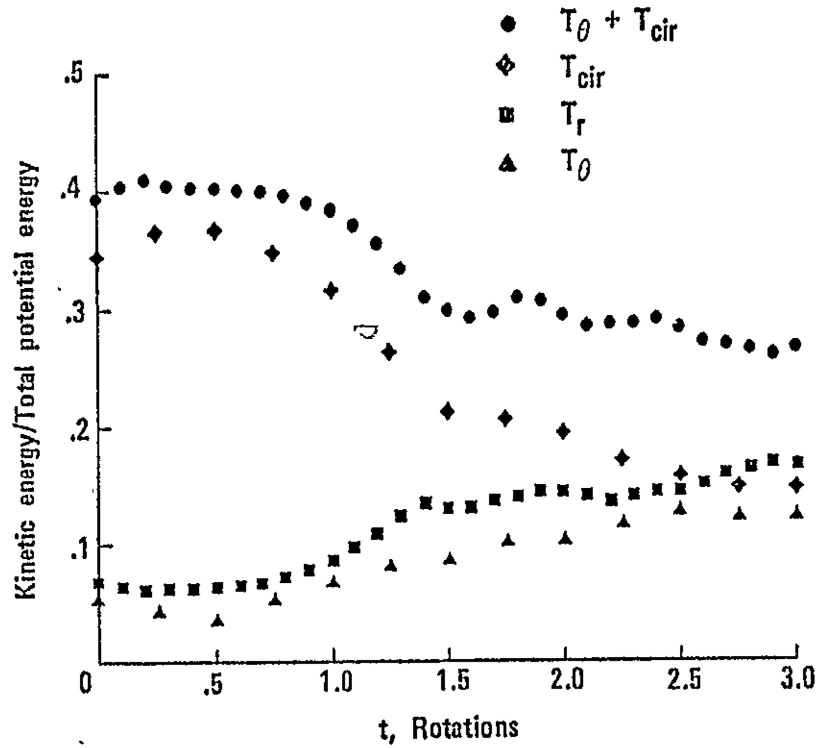

Figure 4.- Time evolution of various kinetic to total potential energy ratios. Note that the ratio of rotational to potential energy is approaching the value of 0.14 predicted by Ostriker and Peebles as required for stability.

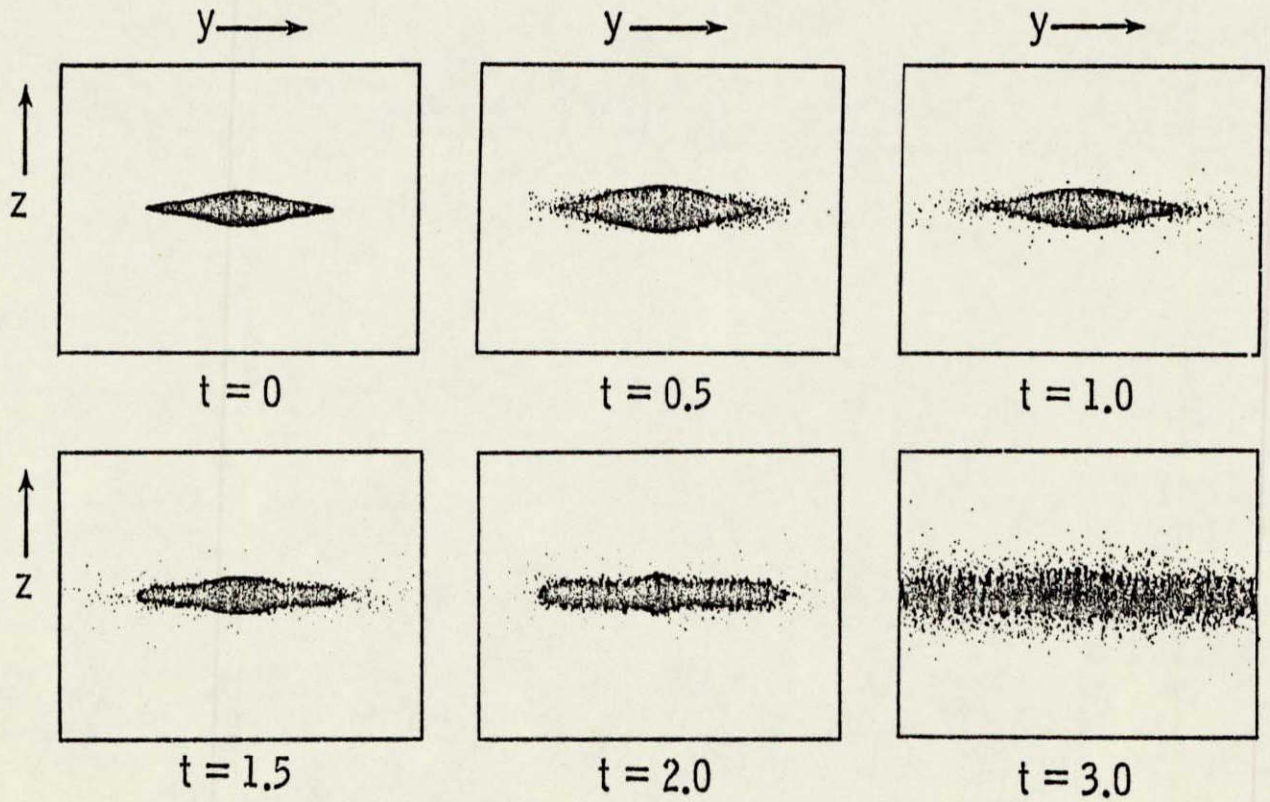

Figure 5.- Side view showing the evolution of the three-dimensional exponential disk. The bar instability results in a rapid expansion in the plane of the disk.

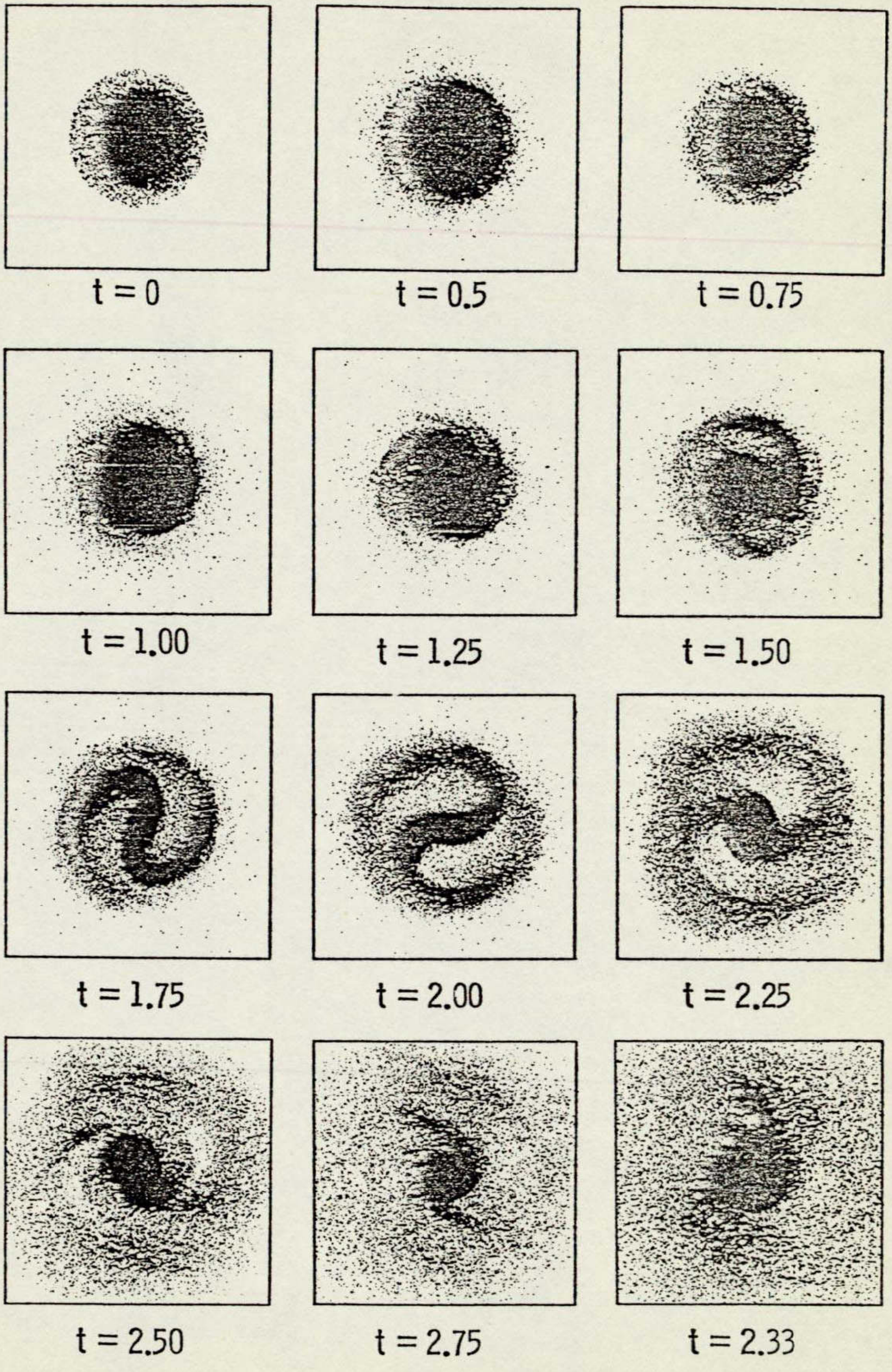

Figure 6.- Evolution of an initially balanced three-dimensional stellar system of 100,000 stars with an exponential radial density variation.

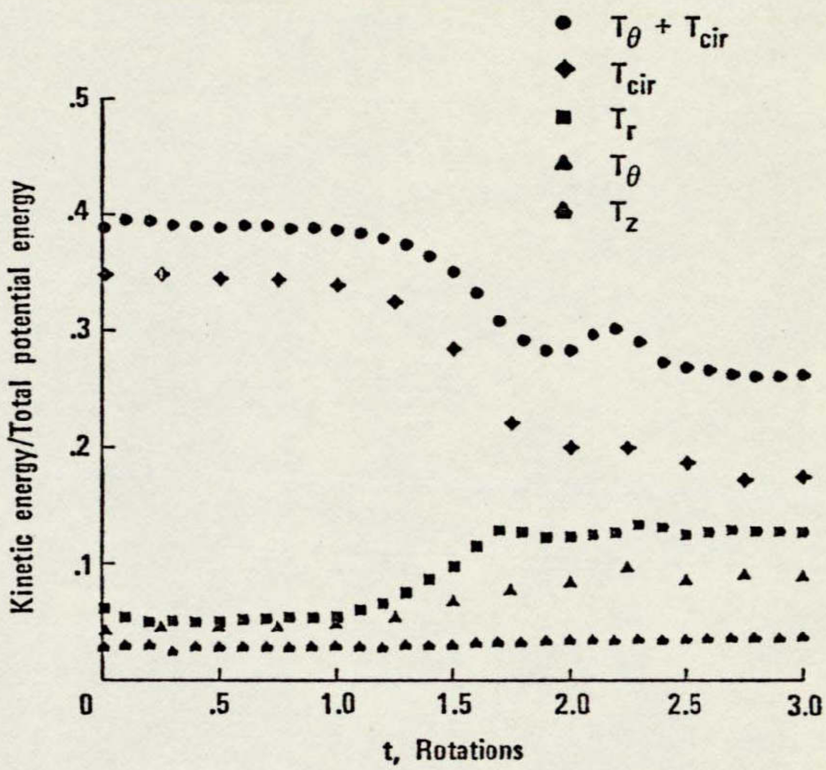

Figure 7.- Time evolution of various kinetic to total potential energy ratios. Note the evolutions of the energy ratios are similar to those shown in figure 4.

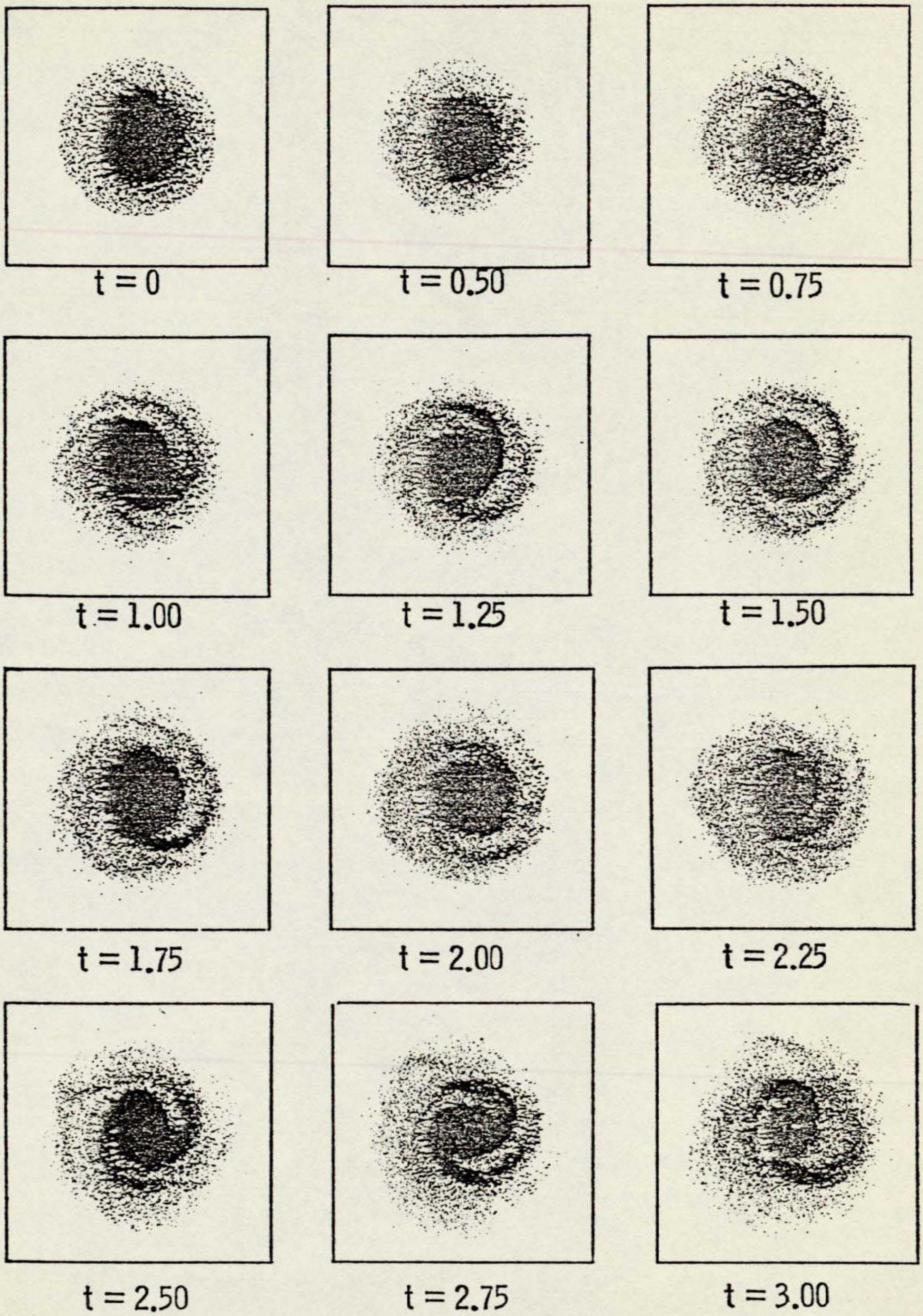

Figure 8.- Evolution of an infinitesimally thin exponential disk with a self-consistent exponential core component. Note that the evolution if considerably less violent than that displayed in figure 1.

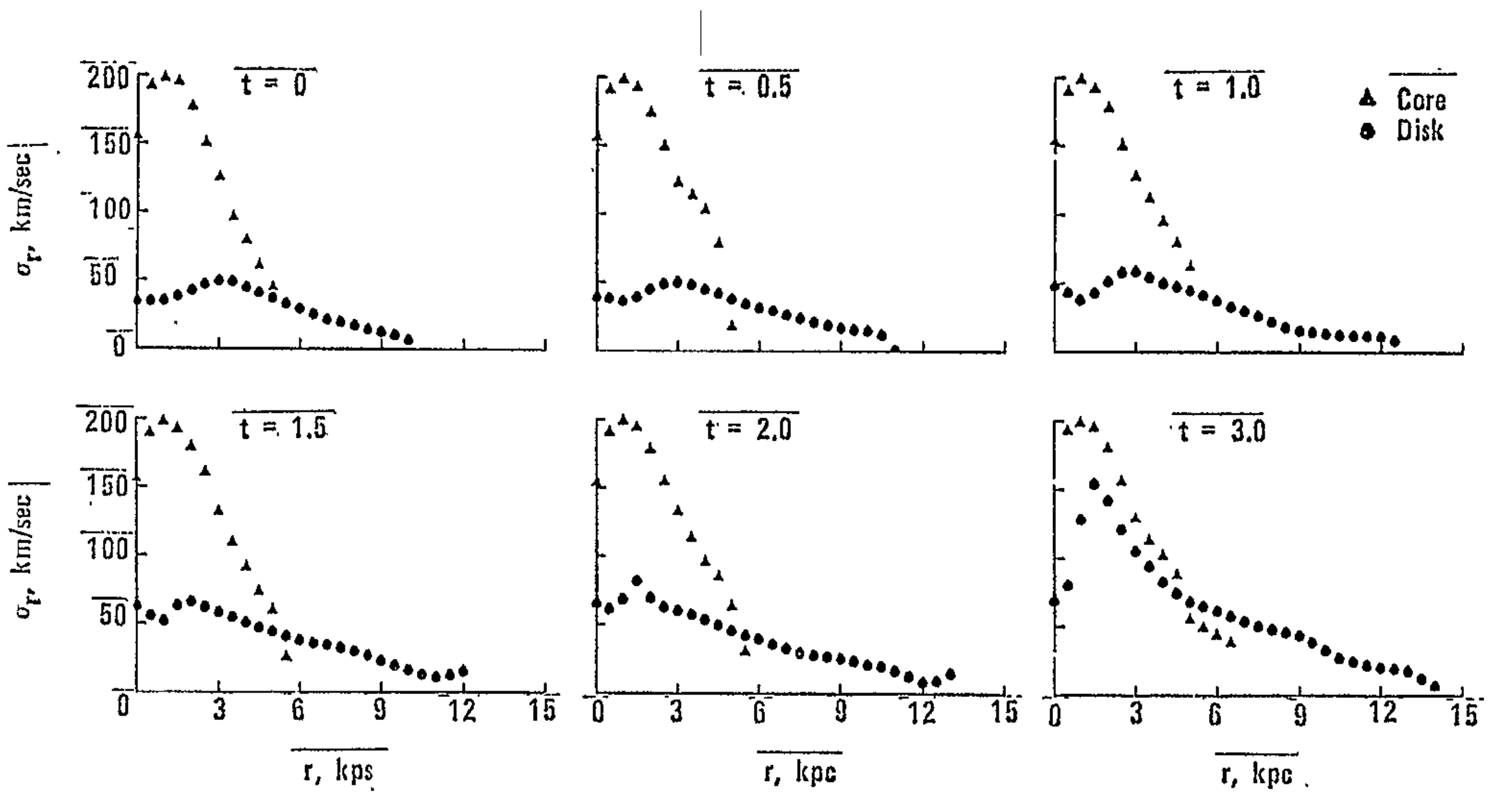

Figure 10.- Evolution of the azimuthally averaged radial velocity dispersion for the two-dimensional exponential disk plus core system.

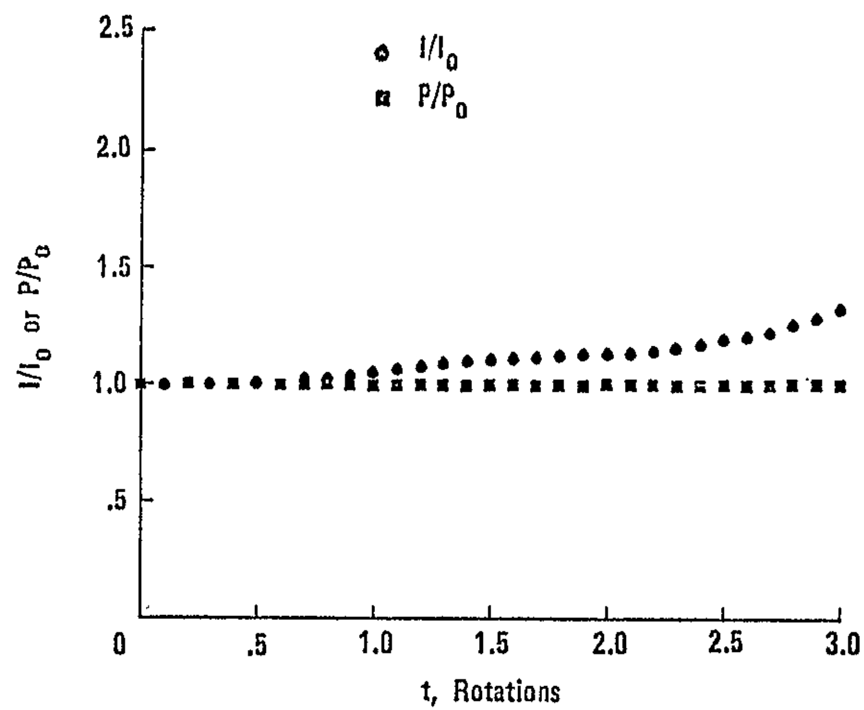

Figure 11.- Time variation of the moment of inertia and the angular momentum for the two-dimensional exponential disk plus core system.

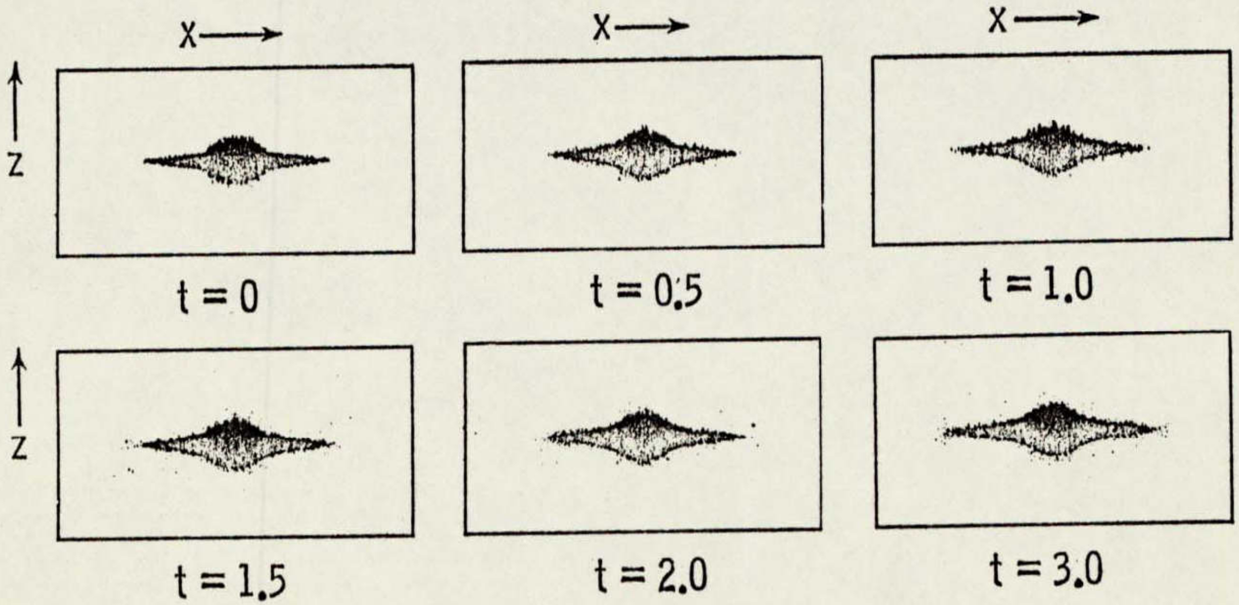

Figure 12.- Side view of the evolution of the three-dimensional exponential disk plus core system. Note the remarkable stability when compared to the three-dimensional disk-only system shown in figure 5.

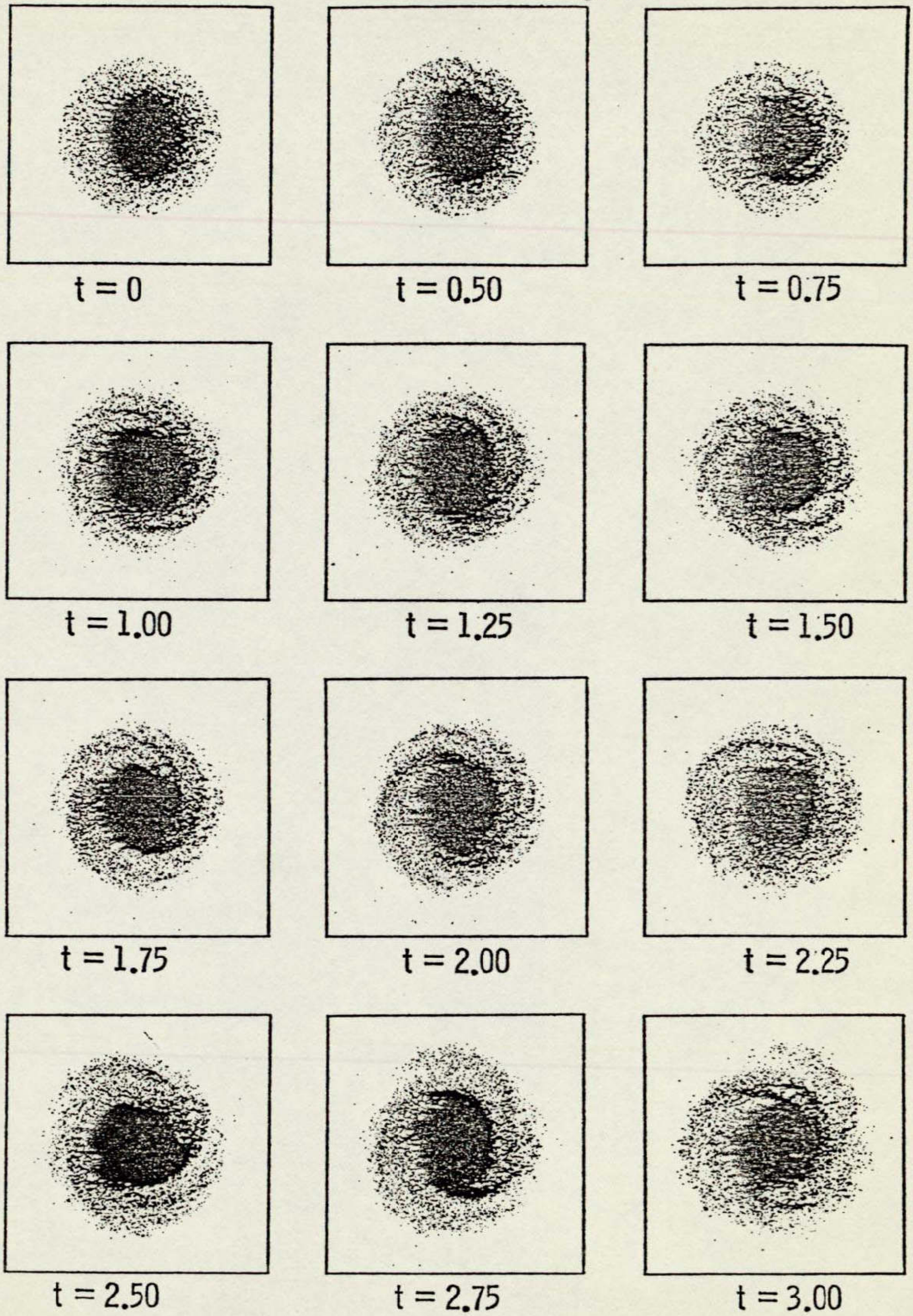

Figure 13.- Evolution of the three-dimensional disk plus core system viewed in the equatorial (x-y) plane. Note the development of the comparably weak spiral structure.

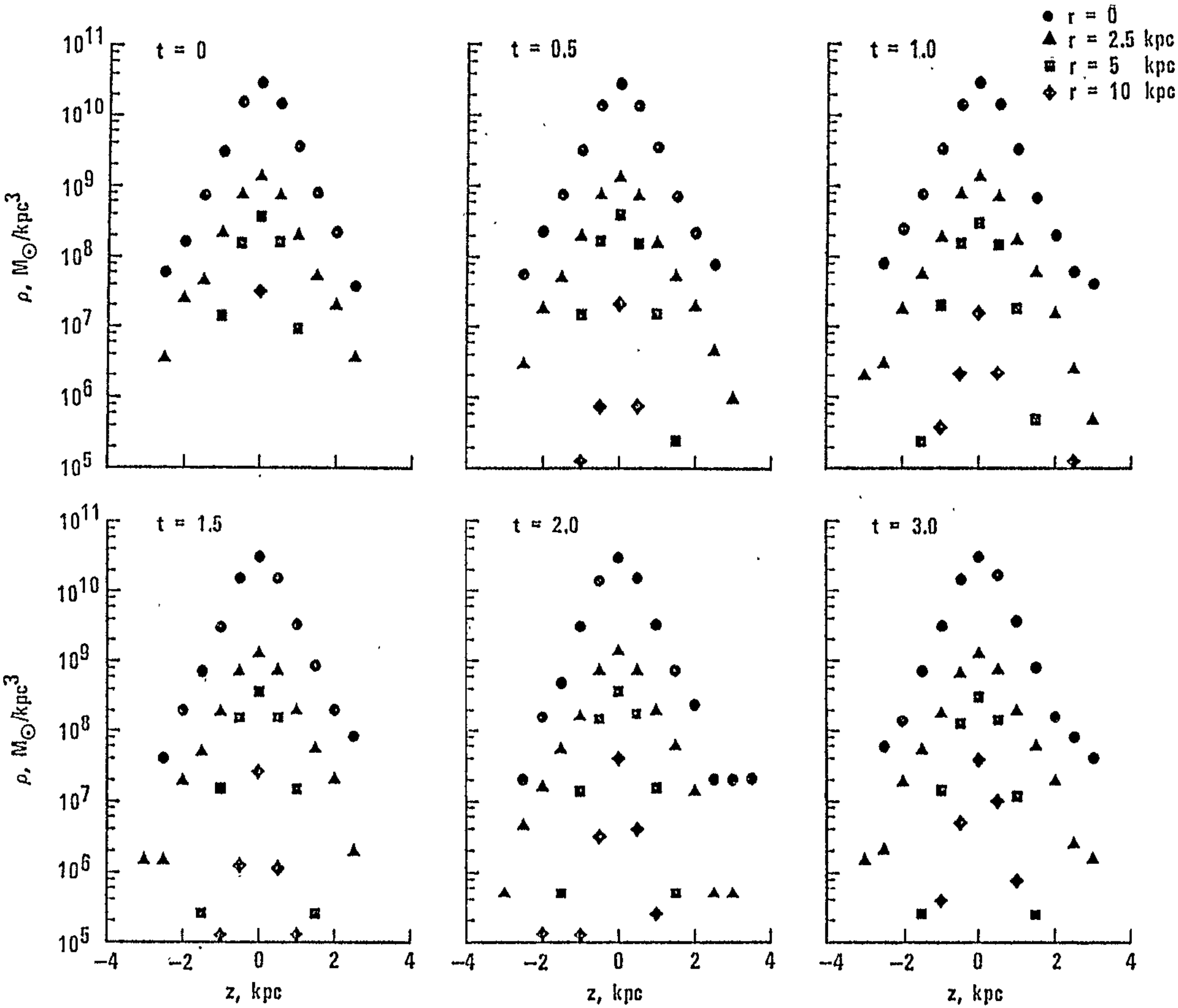

Figure 15.- Evolution of the volume-mass density as a function of z for various radii.

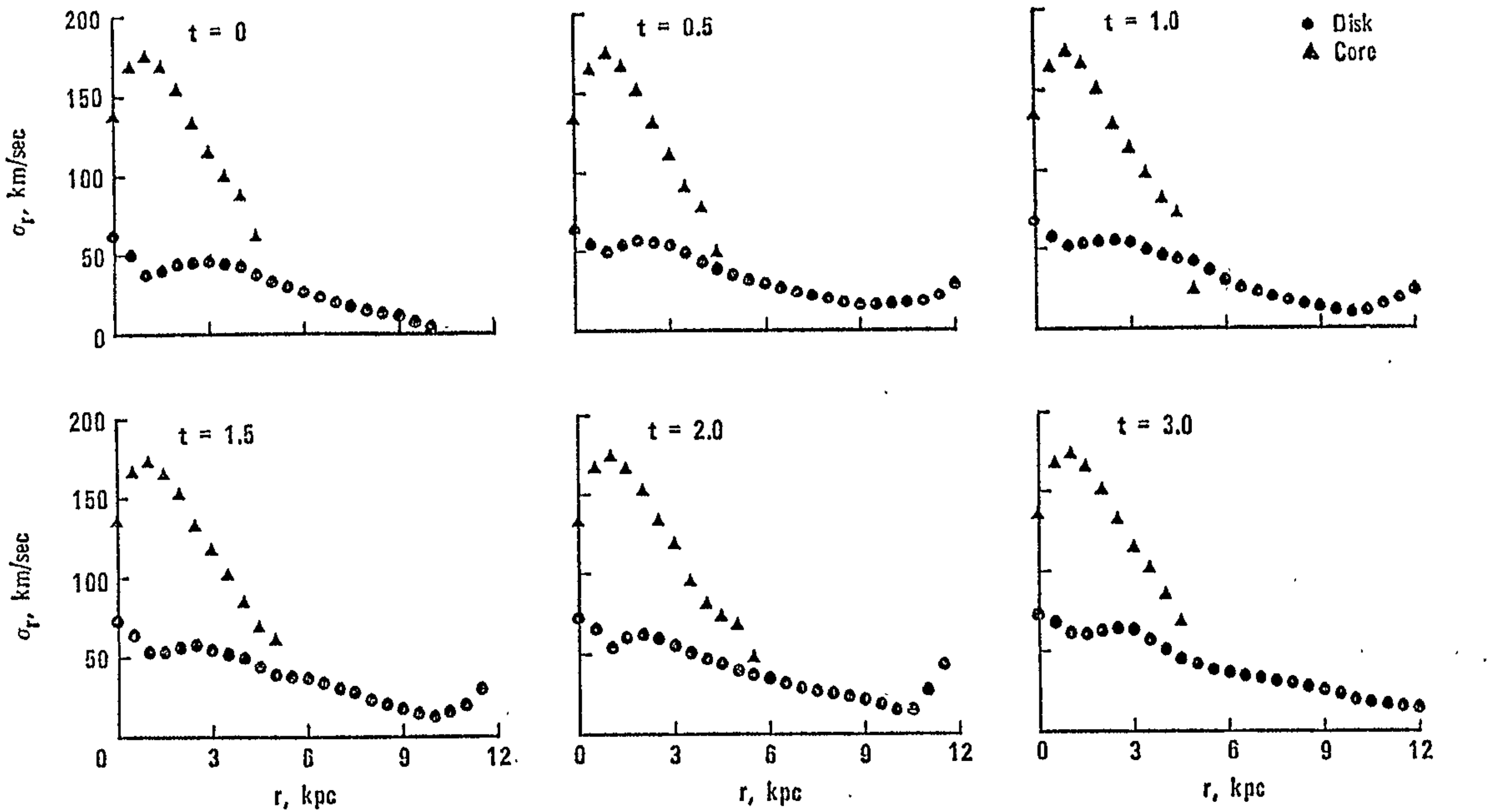

Figure 16.- Evolution of the radial velocity dispersion for the three-dimensional disk plus core system.

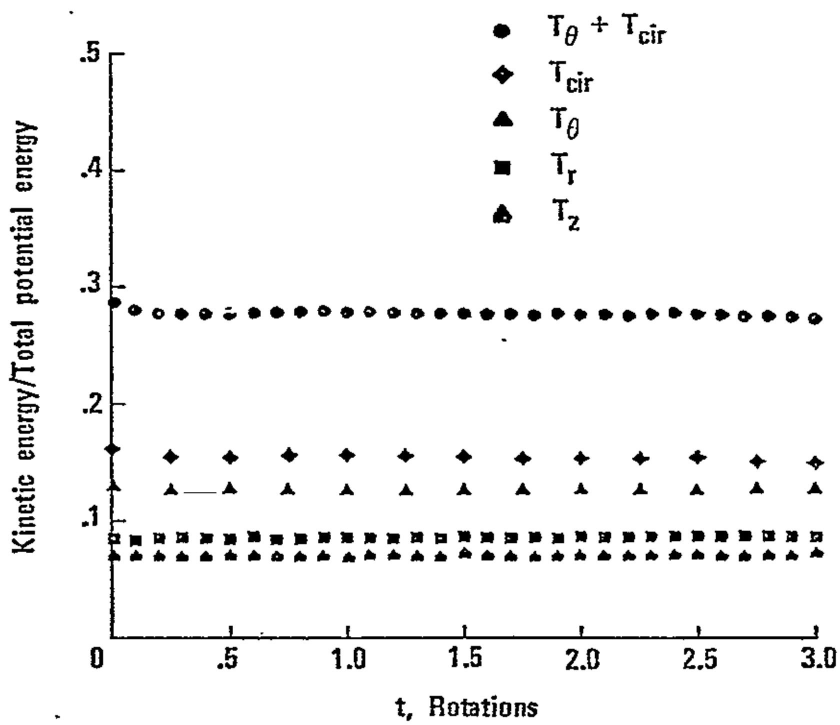

Figure 17.- Evolution of various kinetic to total potential energy ratios for the three-dimensional disk plus core system.

APPENDIX

COMPUTER PROGRAM FOR GENERATING THE THREE-DIMENSIONAL GRAVITATIONAL POTENTIAL DISTRIBUTION OF ISOLATED GALAXIES

MATHEMATICAL SUMMARY

The scaled gravitational potential at the center of cell (x,y,z) is defined by the triple summation over the three-dimensional array of cells

$$\phi_{x,y,z} = \sum_{i=0}^{2n-1} \sum_{j=0}^{2n-1} \sum_{k=0}^{2h-1} \rho_{i,j,k} H_{i-x,j-y,k-z}, \tag{A1}$$

where

$$H_{i,j,k} = (i^2+j^2+k^2)^{-1/2} \quad \text{for} \quad i + j + k \neq 0,$$

$$H_{0,0,0} = 1,$$

and $\rho_{i,j,k}$ is the mass density in cell (i,j,k). Because direct summation is much too time consuming to be practical, the triple summation is evaluated by the convolution method using fast Fourier transforms (ref. (A1)). That is, the Fourier transform of the potential equals the product of the Fourier transforms of ρ and H

$$\tilde{\phi}_{\xi,\eta,\zeta} = \tilde{\rho}_{\xi,\eta,\zeta} \tilde{H}_{\xi,\eta,\zeta} . \tag{A2}$$

The gravitational potential $\phi_{x,y,z}$ is obtained by taking the inverse Fourier transform of equation (A2). Rather than the usual complex Fourier series, here a real expansion is used. For example, the Fourier transform of the density $\rho_{x,y,z}$ is given by

$$\tilde{\rho}_{\xi,\eta,\zeta} = \sum_{z=0}^{2h-1} \sum_{y=0}^{2n-1} \sum_{x=0}^{2n-1} c(x,n)c(y,n)c(z,h)\, \rho_{x,y,z} f(\xi,x,n) f(\eta,y,n) f(\zeta,z,h) \tag{A3}$$

where

$$f(\xi,x,n) = \begin{cases} \cos\ (\xi\ x/n),\ 0 \le \xi \le n \\ \sin\ [\pi(\xi-n)x/n],\ n < \xi < 2n \end{cases}$$

$$c(x,n) = 1/\sqrt{2} \quad \text{if} \quad x = 0 \quad \text{or} \quad x = n,$$

$$c(x,n) = 1,$$

otherwise, the symbols n and h define the $n \times n \times h$ active array and also the $(2n) \times (2n) \times (2h)$ larger array over which the Fourier transform must be taken so that the potential for an isolated galaxy is obtained (see fig. A1). Note that the density may be nonzero only in the smaller $n \times n \times h$ array. Because of the symmetry of $H_{x,y,z}$, the Fourier transform $\hat{\tilde{H}}_{\xi,\eta,\zeta}$ can be obtained by a finite cosine transform

$$\hat{\tilde{H}}_{\xi,\eta,\zeta} = \sum_{z=0}^{h} \sum_{y=0}^{n} \sum_{x=0}^{n} c^2(x,n)\ c^2(y,n)\ c^2(z,h)\ H_{x,y,z}$$

$$\cdot \cos\ (\pi\xi x/n)\ \cos\ (\pi\eta y/n)\ \cos\ (\pi\zeta z/h), \tag{A4}$$

$$0 \le \xi,\eta \le n$$

$$0 \le \xi \le h$$

and

$$\hat{\tilde{H}}_{\xi+n,\eta,\zeta} = \hat{\tilde{H}}_{\xi+n,\eta+n,\zeta} = \hat{\tilde{H}}_{\xi+n,\eta,\zeta+h} = \hat{\tilde{H}}_{\xi+n,\eta+n,\zeta+h}$$

$$= \hat{\tilde{H}}_{\xi,\eta+n,\zeta} = \hat{\tilde{H}}_{\xi,\eta+n,\zeta+h} = \hat{\tilde{H}}_{\xi,\eta,\zeta+h} = \hat{\tilde{H}}_{\xi,\eta,\zeta}.$$

The next step in obtaining the potential is to multiply $\hat{\tilde{\rho}}_{\zeta,\eta,\zeta}$ by $\hat{\tilde{H}}_{\xi,\eta,\zeta}$ to obtain

$$\hat{\tilde{\phi}}_{\xi,\eta,\zeta} = \hat{\tilde{\rho}}_{\xi,\eta,\zeta}\ \hat{\tilde{H}}_{\xi,\eta,\zeta}\ . \tag{A5}$$

The gravitational potential for an isolated galaxy correctly defined over the $n \times n \times h$ aray is obtained by the Fourier synthesis

$$\phi_{x,y,z} = \frac{1}{N^3} \sum_{\zeta=0}^{2h-1} \sum_{\eta=0}^{2n-1} \sum_{\xi=0}^{2n-1} \tilde{\phi}_{\xi,\eta,\zeta} \, (\xi,x,n) f(\eta,y,n) f(\xi,z,h) \tag{A6}$$

Note also, that since

$$\tilde{H}_{\xi,\eta,\zeta} = \tilde{H}_{\xi,\zeta,\eta} = \tilde{H}_{\eta,\xi,\zeta}, \text{ etc.}$$

different permutations of the same set of indices need not be stored. Thus, the transformed Green's function can be converted to a one-dimensional array

$$\tilde{H}_{\xi,\eta,\zeta} = \tilde{F}_n$$

where different permutations of ξ,η,ζ are stored in the same location n given by

$$n = \sum_{i=z}^{\xi} \frac{i}{2}(i-1) + \frac{\eta}{2}(\eta-1) + \zeta$$

$$= \xi(\xi-1)(2\xi-1)/12 + \xi(\xi-1)/4 + \eta(\eta-1)/2 + \zeta$$

Computer Program Subroutine Which Uses Only Core Storage

Table A1 gives a Fortran listing of a computer program which may be used to obtain the potential by use of a (2n) x (2n) x h array of cell. The variables 12A and 13A define the x,y and z dimensions, respectively, of the array used for the potential calculations. When the subroutine GETPHI is called, RHO(I,J,K) contains the mass density and GETPHI places the values of the corresponding gravitational potential in RHO(I,J,K). The subroutine FTRANS(I,I2B) has been written by R. Hockney (ref. A2) and it performs a finite Fourier analysis or synthesis on the common input array Z and places the result in the common output array Y. The subroutine performs a cosine analysis for I = 2, a periodic analysis for I = 3, and a periodic synthesis for I = 4. The subroutine GETSET(I,I2B) initializes FTRANS and is called every time the arguments of FTRANS(I,I2B) are changed. The Fourier transform $\tilde{H}_{\xi,\eta,\zeta}$ is calculated on an $(n+1) \times (n+1) \times (h+1)$ array only the first time that the subroutine is called and is kept in storage for subsequent use.

The Fourier transform of $\rho_{x,y,z}$ in the x-direction is generated by obtaining the partial transform $\tilde{\rho}_{\xi,y,z}$ for $0 \leq \xi \leq 2n-1$, $0 \leq y \leq n-1$ and $0 \leq z \leq h-1$. $\tilde{\rho}_{\xi,y,z}$ is zero outside of this region because $\rho_{x,y,z}$ is nonzero only over the $n \times n \times h$ active array. Next, the Fourier transform of $\tilde{\rho}_{\xi,y,z}$ is performed in the y-direction obtaining the x-y partial transform $\tilde{\rho}_{\xi,h,z}$ for $0 \leq \xi \leq 2n-1$, $0 \leq \eta \leq 2n-1$ and $0 \leq z \leq h-1$. Since $\tilde{\rho}_{\xi,\eta,z}$ is zero for $h \leq z \leq 2h-1$, by use of one-dimensional arrays Y and Z the Fourier transform of $\tilde{\rho}_{\xi,\eta,z}$ can be taken in the z-direction to obtain the total transform $\tilde{\rho}_{\xi,\eta,\zeta}$ for $0 \leq \zeta \leq 2h-1$. Next, $\tilde{\rho}_{\xi,\eta,\zeta}$ is multiplied by $\tilde{H}_{\xi,\eta,\zeta}$ to obtain $\tilde{\phi}_{\xi,\eta,\zeta}$ and the inverse Fourier transform is performed in the z-direction. The resulting partial x-y transform $\tilde{\phi}_{\xi,\eta,}$ is placed in the $2n \times 2n \times h$ RHO(I,J,K) array for $0 \leq \xi \leq 2n-1$, $0 \leq \eta \leq 2n-1$ and $0 \leq z \leq h-1$ with values for $h \leq z \leq 2h-1$ discarded. (The use of these one-dimensional arrays was first presented in reference A3 for a two-dimensional potential solver). Next, the inverse Fourier transform of $\tilde{\phi}_{\xi,\eta,z}$ is generated in the y-direction by obtaining the x-partial transform $\tilde{\phi}_{\xi,y,z}$ for $0 \le \xi \le 2n-1$, $0 \le y \le n-1$ and $0 \le z \le h-1$. The final step is to perform the inverse Fourier transform in the x-direction for $0 \le y \le n-1$ and $0 \le z \le h-1$ to yield the correct gravitational potential $\phi_{x,y,z}$ for an isolated galaxy over the $n \times n \times h$ array.

Overlayed Computer Program Which Uses Core and Disk Storage

The use of the listing of Table A1 with the 64 x 64 x 16 active density/potential array used in this paper would have necessitated the dimensioning of the RHO array at 128 x 128 x 16 and the H array at 65 x 65 x 17. As such, large dimensions would have excluded use of the CDC 6600 computer, the listing of Table A1 was modified to include use of overlayed programs and disk storage resulting in a maximum core storage at any one time of array elements equaling about five fourths of the active array. The listing of this program in Table A2 includes (a) a section of an initializing overlay in which relevant constants are computed (b) a section of the star advancing overlay in which "chunks" of the density array are written on appropriate disk files, (c) another section of the star advancing overlay in which "chunks" of the computed potential array are read from disk files, (d) the GETH overlay which computer $\tilde{H}$, and (e) the GETPHI overlay which computes the potential array from the density array.

The method used is the alignment in the direction of transformation of four identical arrays named RHO1, RHO2, RHO3, and RHO4, each of which is dimensioned $(n/2) \times (n/2) \times h$ within the GETPHI overlay. (See figs. A2 and A5. For clarity, figures A1 through A6 are drawn for an active array dimensioned $n \times n \times h = 8 \times 8 \times 4$; table A3 compares the array dimensions of these figures and the listing of table A2.) The active array is dimensioned as the PHI array within the initializing and star advancing overlays (see figures A1 and A3) but is not dimensioned within the GETPHI overlay. As figure A2 suggests, the "chunks" RHO1, RHO2, RHO3 and RHO4 may be visualized as forming either a row or a column of the lower half $(0 \leq z \leq h-1)$ of the extended array. Switching the lineup to a different row or column is accomplished by storing the array associated with each "chunk" location on a separate file; these eight files are also indicated in figure A2.

As shown in figure A3 one "chunk" size array named OI is dimensioned in the initializing and star advancing overlays. "Chunks" of the active array are transferred between the PHI array of these overlays and the arrays RHO1, RHO2, RHO3 and RHO4 of the GETPHI overlay via "do loop" transfer to/from the OK array and storage on files 1, 2, 5 and 6.

At the beginning of a program run, the GETH overlay computes $\tilde{H}$ in the $(n+1) \times (n+1) \times (h+1)$ H array in the same manner as the listing of table A1. All of $\tilde{H}_{\xi,\eta,\zeta}$, except for two boundary planes of elements $(\xi = n, 0 \leq \eta \leq n, 0 \leq \zeta \leq h$ and $0 \leq \xi \leq n, \eta = n, 0 \leq \zeta \leq h)$, is then transferred in portions via "do loop" to the $(n/2) \times (n/2) \times (h+1)$ HH array from which it is written on disk file 9 (see figure A4). Elements of one boundary plane of $\tilde{H}_{\xi,\eta,\zeta}$ $(\xi = n, 0 \leq \eta \leq n, 0 \leq \zeta \leq h)$ are transferred to the $(n+1) \times (h+1)$ HN21 array which is in common with the GETPHI overlay; the ζ-η transpose of that boundary plane is equal to the other boundary plane $(0 \leq \zeta \leq n,\ \eta = n,\ 0 \leq \zeta \leq h)$ due to the symmetry of $\tilde{H}$ across the $\zeta=\eta$ diagonal plane. During each potential solution the portions of $\tilde{H}$ on file 9 are read sequentially into an $(n/2) \times (n/2) \times (h+1)$ HH array of the GETPHI overlay from which $\tilde{H}$ elements, along with those in the HN21 array, are multiplied with $\tilde{\rho}$. This sequence (listed in table A4) utilizes the symmetry and periodicity of $\tilde{H}$ (equation (A4)) to provide a full set of $(2n) \times (2n) \times (2h)$ $\tilde{H}$ elements to the GETPHI overlay in a manner which minimizes the reading of file 9.

The GETPHI overlay consists of subroutines ANLX(JCOLUMN), ANLSYN(IROW) and SYNX(JCOLUMN) which dimension in common the arrays HH, HN21, RHO1, RHO2, RHO3 and RHO4 as pictures in figure 5. Figure 6 indicates the lineup of "chunks" associated with each call to a subroutine. The potential solution is mathematically identical with that described for the listing of table A1. Calling ANLX(1) and ANLX(2) performs the Fourier transform of $\rho_{x,y,z}$ in the x-direction to form $\tilde{\rho}_{\zeta,y,z}$. Calling ANLSYN(1), ANLSYN(3), ANLSYN(2) and ANLSYN(4) in sequence performs the following: (a) a Fourier transform of $\tilde{\rho}_{\zeta,y,z}$ in the y- and z-directions to form $\tilde{\rho}_{\xi,\eta,\zeta}$; (b) multiplication with $\tilde{H}_{\xi,\eta,\zeta}$ to form $\tilde{\phi}_{\xi,\eta,\zeta}$; and (c) the inverse Fourier transform of $\tilde{\phi}_{\xi,\eta,\zeta}$ in the z- and y-directions to form $\tilde{\phi}_{\zeta,y,z}$. Calling SYNX(1) and SYNX(2) performs the inverse Fourier transform of $\tilde{\phi}_{\xi,y,z}$ in the x-direction to form $\phi_{x,y,z}$. The GETPHI overlay is outlined in more detail in table A5.

Efficiencies of the Two Computer Programs

The program of table A2 is considerably more efficient than that of able A1 because the addition of some peripheral processing time and a nall increase in central processing time is much more than compensated or by a 75 percent decrease in the required core storage. The maximum umber of active array elements dimensionable on the CDC 6600 with the rograms of table A1 and A2 are respectively 16384 (e.g. 32 x 32 x 16) nd 65536 (e.g. 64 x 64 x 16); the latter program can have other otentially useful active array dimensions of 32 x 32 x 8, 32 x 32 x 16, nd 32 x 32 x 32. Solution of the 64 x 64 x 16 active array by the DC 6600 requires about 300 (octal) words of core storage and with $\tilde{H}$ lready computed takes about 75 seconds of central processing time.

REFERENCES

A1. Hohl, Frank; and Hockney, R. W.: A Computer Model of Disks of Stars. J. Comput. Phys., vol. 4, no. 3, Oct. 1969, pp. 306-324.

A2. Hockney, R. W.: The Potential Calculation and Some Applications. Methods in Computational Physics, vol. 9 - Plasma Physics, Berni Alder; Sidney Fernbach; and Mannuel Rotenberg, eds., Academic Press, 1970, pp. 135-211.

A3. Hohl, Frank: Evolution of a Stationary Disk of Stars. J. Comput. Phys., vol. 9, no. 1, Feb. 1972, pp. 10-25.

TABLE A1

SUBROUTINE FOR CALCULATING THE THREE-DIMENSIONAL GRAVITATIONAL POTENTIAL USING ONLY CORE STORAGE

```
      SUBROUTINE GETPHI
      COMMON Z(1025),Y(1025),RHO(64,64,16),I2A,I3A,ITEST
      DIMENSION H(33,33,17)
      IF(ITEST.EQ.0) GO TO 11
      ITEST=0
      I2B=I2A-1
      N=2**I2A
      NO2=N/2
      N21=NO2+1
      I3B=I3A-1
      NH=2**I3A
      NHO2=NH/2
      NH21=NHO2+1
      RNI=1./(N*N*NH)
      DO 1 K=1,NH21
      DO 1 J=1,N21
      DO 1 I=1,N21
      RI=(K-1)*(K-1)+(J-1)*(J-1)+(I-1)*(I-1)
      IF(RI.LT.1.) RI=1.
      H(I,J,K)=RNI/SQRT(RI)
    1 CONTINUE
      CALL GETSET(2,I2B)
      DO 2 K=1,NH21
      DO 2 J=1,N21
      DO 3 I=1,N21
    3 Z(I)=H(I,J,K)
      CALL FTRANS(2,I2B)
      DO 4 I=1,N21
    4 H(I,J,K)=Y(I)
    2 CONTINUE
      DO 5 K=1,NH21
      DO 5 I=1,N21
      DO 6 J=1,N21
    6 Z(J)=H(I,J,K)
      CALL FTRANS(2,I2B)
      DO 7 J=1,N21
    7 H(I,J,K)=Y(J)
    5 CONTINUE
      CALL GETSET(2,I3B)
      DO 10 J=1,N21
      DO 10 I=1,N21
      DO 8 K=1,NH21
    8 Z(K)=H(I,J,K)
      CALL FTRANS(2,I3B)
      DO 9 K=1,NH21
    9 H(I,J,K)=Y(K)
   10 CONTINUE
   11 CONTINUE
      WRITE(6,43)
   43 FORMAT(10H  H(I,J,K))
      DO  42 K=1,NH21
      DO 42 J=1,N21
      WRITE(6,41) J,K
      WRITE(6,40) (H(I,J,K),I=1,N21)
   41 FORMAT(14H  I=1,N21    J=I3,5H    K=I3)
   40 FORMAT(2H  8E16.8)
   42 CONTINUE
      CALL GETSET(3,I2A)
      DO 14 K=1,NHO2
      DO 14 J=1,NO2
      DO 12 I=1,N
   12 Z(I)=RHO(I,J,K)
      CALL FTRANS(3,I2A)
      DO 13 I=1,N
   13 RHO(I,J,K)=Y(I)
   14 CONTINUE
```

```
      DO 17 K=1,NHO2
      DO 17 I=1,N
      DO 15 J=1,N
   15 Z(J)=RHO(I,J,K)
      CALL FTRANS(3,I2A)
      DO 16 J=1,N
   16 RHO(I,J,K)=Y(J)
   17 CONTINUE
      DO 20 I=1,N
      DO 20 J=1,N
      DO 18 K=1,NHO2
      Z(K)=RHO(I,J,K)
   18 Z(K+NHO2)=0.
      CALL GETSET(3,I3A)
      CALL FTRANS(3,I3A)
      IF(I.GT.N21.AND.J.LE.N21) GO TO 22
      IF(I.LE.N21.AND.J.GT.N21) GO TO 24
      IF(I.GT.N21.AND.J.GT.N21) GO TO 26
      DO 19 K=1,NHO2
      Z(K)=Y(K)*H(I,J,K)
   19 Z(K+NHO2)=Y(K+NHO2)*H(I,J,K)
      Z(1)=Y(1)*H(I,J,1)
      Z(NH21)=Y(NH21)*H(I,J,NH21)
      GO TO 21
   22 DO 23 K=2,NHO2
      Z(K)=Y(K)*H(I-NO2,J,K)
   23 Z(K+NHO2)=Y(K+NHO2)*H(I-NO2,J,K)
      Z(1)=Y(1)*H(I-NO2,J,1)
      Z(NH21)=Y(NH21)*H(I-NO2,J,NH21)
      GO TO 21
   24 DO 25 K=2,NHO2
      Z(K)=Y(K)*H(I,J-NO2,K)
   25 Z(K+NHO2)=Y(K+NHO2)*H(I,J-NO2,K)
      Z(1)=Y(1)*H(I,J-NO2,1)
      Z(NH21)=Y(NH21)*H(I,J-NO2,NH21)
      GO TO 21
   26 DO 27 K=2,NHO2
      Z(K)=Y(K)*H(I-NO2,J-NO2,K)
   27 Z(K+NHO2)=Y(K+NHO2)*H(I-NO2,J-NO2,K)
      Z(1)=Y(1)*H(I-NO2,J-NO2,1)
      Z(NH21)=Y(NH21)*H(I-NO2,J-NO2,NH21)
   21 CONTINUE
      CALL GETSET(4,I3A)
      CALL FTRANS(4,I3A)
      DO  28 K=1,NHO2
   28 RHO(I,J,K)=Y(K)
   20 CONTINUE
      CALL GETSET(4,I2A)
      DO 29 K=1,NHO2
      DO 29 J=1,N
      DO 30 I=1,N
   30 Z(I)=RHO(I,J,K)
      CALL FTRANS(4,I2A)
      DO 31 I=1,N
   31 RHO(I,J,K)=Y(I)
   29 CONTINUE
      DO 32 K=1,NHO2
      DO 32 I=1,NO2
      DO 33 J=1,N
   33 Z(J)=RHO(I,J,K)
      CALL FTRANS(4,I2A)
      DO 34 J=1,NO2
   34 RHO(I,J,K)=Y(J)
   32 CONTINUE
      RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE A2

OVERLAYS FOR CALCULATING THE THREE-DIMENSIONAL GRAVITATIONAL POTENTIAL USING CORE AND DISK STORAGE

```
C THE FOLLOWING IS THE SECTION OF AN INITIALIZING OVERLAY IN WHICH CONSTANTS
C RELATED TO THE DIMENSIONS OF THE PHI (DENSITY/POTENTIAL) ARRAY ARE COM-
C PUTED.  IT IS CALLED ONCE AT THE BEGINNING OF A PROGRAM RUN.  IN THIS
C LISTING THE VALUES OF I2A, I3A AND THE DIMENSION AND LABELED COMMON
C STATEMENTS ARE SET FOR AN ACTIVE PHI ARRAY DIMENSIONED 64 BY 64 BY 16.
      I2A=7
      I3A=5
      I2B=I2A-1
      I3B=I3A-1
      N=2**I2A
      NO2=N/2
      N21=NO2+1
      NO4=N/4
      N34=NO2+NO4
      NH=2**I3A
      NHO2=NH/2
      NH21=NHO2+1
C
C
C
C ****************************************************************************
C ****************************************************************************
C THE FOLLOWING IS THE SECTION OF THE STAR ADVANCING OVERLAY IN WHICH CHUNKS
C OF THE PHI ARRAY (CONTAINING THE DENSITY MESH) ARE WRITTEN ONTO DISK FILES
C 1,2,5 AND 6.  THE STAR ADVANCING OVERLAY IS CALLED ONCE PER TIME STEP.
      DIMENSION PHI(64,64,16),OI(32,32,16)
      DO 520 K=1,NHO2
      DO 520 J=1,NO4
      DO 520 I=1,NO4
  520 OI(I,J,K)=PHI(I,J,K)
      WRITE(1) OI
      REWIND 1
      DO 525 K=1,NHO2
      DO 525 J=1,NO4
      DO 525 I=1,NO4
  525 OI(I,J,K)=PHI(I,NO4+J,K)
      WRITE(5) OI
      REWIND 5
      DO 530 K=1,NHO2
      DO 530 J=1,NO4
      DO 530 I=1,NO4
  530 OI(I,J,K)=PHI(NO4+I,J,K)
      WRITE(2) OI
      REWIND 2
      DO 535 K=1,NHO2
      DO 535 J=1,NO4
      DO 535 I=1,NO4
  535 OI(I,J,K)=PHI(NO4+I,NO4+J,K)
      WRITE(6) OI
      REWIND 6
C
C
C
C****************************************************************************
C****************************************************************************
C THE FOLLOWING IS THE SECTION OF THE STAR ADVANCING OVERLAY IN WHICH CHUNKS
C OF THE PHI ARRAY (CONTAINING THE POTENTIAL MESH) ARE READ FROM DISK FILES
C 1,2,5 AND 6.
```

```
      DIMENSION PHI(64,64,16),OI(32,32,16)
      READ(1) OI
      REWIND 1
      DO 30 K=1,NHO2
      DO 30 J=1,NO4
      DO 30 I=1,NO4
   30 PHI(I,J,K)=OI(I,J,K)
      READ(5) OI
      REWIND 5
      DO 40 K=1,NHO2
      DO 40 J=1,NO4
      DO 40 I=1,NO4
   40 PHI(I,NO4+J,K)=OI(I,J,K)
      READ(2) OI
      REWIND 2
      DO 50 K=1,NHO2
      DO 50 J=1,NO4
      DO 50 I=1,NO4
   50 PHI(NO4+I,J,K)=OI(I,J,K)
      READ(6) OI
      REWIND 6
      DO 60 K=1,NHO2
      DO 60 J=1,NO4
      DO 60 I=1,NO4
   60 PHI(NO4+I,NO4+J,K)=OI(I,J,K)
C
C
C
C***********************************************************************
C***********************************************************************
C THE FOLLOWING IS THE GETH OVERLAY,  WHICH COMPUTES AND STORES THE TRANS-
C FORMED GREENS FUNCTION.  IT IS CALLED ONCE AT THE BEGINNING OF A PROGRAM
C RUN.
      OVERLAY(IFILE,4,0)
      PROGRAM GETH
C THIS OVERLAY PERFORMS A COSINE ANALYSIS OF THE THREE-DIMENSIONAL GREENS
C FUNCTION ARRAY. IT THEN WRITES CHUNKS OF THIS ARRAY ON DISK FILE 9 IN THE
C ORDER IN WHICH THEY WILL BE READ INTO THE HH ARRAY DURING THE GETPHI
C OVERLAY. VALUES FOR I=N/2+1 AND J=N/2+1 ARE TRANSFERRED TO THE HN21 ARRAY
C WHICH IS IN COMMON WITH THE GETPHI OVERLAY.
      COMMON/ALLCOM/N,NO2,N21,NO4,N34,NH,NHO2,NH21,I2A,I2B,I3A,I3B
      COMMON/HN21COM/HN21(65,17)
      COMMON Z(1025), Y(1025)
      DIMENSION H(65,65,17),HH(32,32,17)
      RNI=1./(N*N*NH)
      DO 1 K=1,NH21
      DO 1 J=1,N21
      DO 1 I=1,N21
      RI=(K-1)*(K-1)+(J-1)*(J-1)+(I-1)*(I-1)
      IF(RI.LT.1.) RI=1.
      H(I,J,K)=RNI/SQRT(RI)
    1 CONTINUE
      CALL GETSET(2,I2B)
      DO 2 K=1,NH21
      DO 2 J=1,N21
      DO 3 I=1,N21
    3 Z(I)=H(I,J,K)
```

```
      CALL FTRANS(2,128)
      DO 4 I=1,N21
    4 H(I,J,K)=Y(I)
    2 CONTINUE
      DO 5 K=1,NH21
      DO 5 I=1,N21
      DO 6 J=1,N21
    6 Z(J)=H(I,J,K)
      CALL FTRANS(2,128)
      DO 7 J=1,N21
    7 H(I,J,K)=Y(J)
    5 CONTINUE
      CALL GETSET(2,138)
      DO 10 J=1,N21
      DO 10 I=1,N21
      DO 8 K=1,NH21
    8 Z(K)=H(I,J,K)
      CALL FTRANS(2,138)
      DO 9 K=1,NH21
    9 H(I,J,K)=Y(K)
   10 CONTINUE
      DO 30 I=1,NO4
      DO 30 J=1,NO4
      DO 30 K=1,NH21
   30 HH(I,J,K)=H(I,J,K)
      WRITE(9) HH
      DO 35 I=1,NO4
      DO 35 J=1,NO4
      DO 35 K=1,NH21
   35 HH(I,J,K)=H(I,NO4+J,K)
      WRITE(9) HH
      DO 40 I=1,NO4
      DO 40 J=1,NO4
      DO 40 K=1,NH21
   40 HH(I,J,K)=H(I,J,K)
      WRITE(9) HH
      DO 45 I=1,NO4
      DO 45 J=1,NO4
      DO 45 K=1,NH21
   45 HH(I,J,K)=H(NO4+I,J,K)
      WRITE(9) HH
      DO 50 I=1,NO4
      DO 50 J=1,NO4
      DO 50 K=1,NH21
   50 HH(I,J,K)=H(NO4+I,NO4+J,K)
      WRITE(9) HH
      DO 55 I=1,NO4
      DO 55 J=1,NO4
      DO 55 K=1,NH21
   55 HH(I,J,K)=H(NO4+I,J,K)
      WRITE(9) HH
      REWIND 9
      DO 15 K=1,NH21
      DO 15 I=1,N21
   15 HN21(I,K)=H(I,N21,K)
      RETURN
      END
```

```
C
C
C
C*************************************************************************
C*************************************************************************
C THE FOLLOWING IS THE GETPHI OVERLAY, WHICH COMPUTES THE POTENTIAL MESH.
C IT REPLACES CHUNKS OF DENSITY STORED ON DISK FILES 1,2,5 AND 6 WITH
C CORRESPONDING CHUNKS OF THE POTENTIAL MESH.  IT IS CALLED ONCE PER TIME
C STEP.
      OVERLAY(GFILE,5,0)
      PROGRAM GETPHI
C THIS OVERLAY SOLVES FOR THE POTENTIAL MESH (DIMENSIONED N/2 BY N/2 BY
C NH/2) DUE TO A DENSITY MESH (DIMENSIONED N/2 BY N/2 BY NH/2) BY DOING A
C PERIODIC ANALYSIS OF THE DENSITY AND THEN A PERIODIC SYNTHESIS OF THE
C PRODUCT OF THE TRANSFORMED GREENS FUNCTION (DIMENSIONED (N/2+1) BY (N/2+1)
C BY (NH/2+1)) AND THE TRANSFORMED DENSITY.  FORMALLY SPEAKING, EACH OF THE
C TRANSFORMS (EXCEPT THE COSINE ANALYSIS OF THE GREENS FUNCTION, WHICH IS
C PERFORMED IN THE GETH OVERLAY) REQUIRES AN ARRAY DIMENSIONED N BY N BY
C NH.  TO REDUCE CORE STORAGE THIS OVERLAY PERFORMS THESE TRANSFORMS IN
C CHUNKS BY THE ALIGNMENT OF FOUR SMALLER ARRAYS NAMED RHO1, RHO2, RHO3, AND
C RHO4, EACH OF WHICH IS DIMENSIONED N/4 BY N/4 BY NH/2.  THE CHUNKS OF THE
C LOWER HALF (1 .LE. Z .LE. NH/2) OF THE EXTENDED ARRAY NOT IN CORE AT ANY
C ONE TIME ARE STORED ON DISK FILES 1 THROUGH 8.  THE FOLLOWING ARE TWO TOP
C VIEWS OF THE LOWER HALF OF THE EXTENDED ARRAY.  BOTH OF THESE VIEWS
C DESIGNATE THE CHUNKS AS IROW AND JCOLUMN.   IROW 1 AND 2 OF JCOLUMN
C 1 AND 2 CONSTITUTE THE ACTIVE MESH.  IN THE DIAGRAM ON THE LEFT THE
C NUMBERS WITHIN THE CHUNKS OF JCOLUMN 1 AND 2 INDICATE THE DISK FILES ON
C WHICH THOSE CHUNKS ARE STORED. (NO DISK STORAGE IS REQUIRED FOR JCOLUMN 3
C OR 4.)  REFERRING TO THE DIAGRAM ON THE RIGHT, THE NUMBERS WITHIN THE
C CHUNKS ARE THE ORDER IN WHICH CHUNKS OF THE TRANSFORMED DENSITY ARE
C MUTIPLIED (ELEMENT BY ELEMENT) BY THE APPROPRIATE PORTION OF THE
C TRANSFORMED GREENS FUNCTION WHICH HAS BEEN READ FROM DISK FILE 9 INTO
C ARRAY HH(N/4,N/4,NH/2+1).  (AN EXCEPTION IS THE SET OF TRANSFORMED GREENS
C FUNCTION BOUNDARY VALUES FOR I=N/2+1 AND J=N/2+1 WHICH REMAIN AT ALL TIMES
C IN COMMON IN THE ARRAY HN21(N/2+1,NH/2+1).)  A PLUS IN A CHUNK INDICATES
C THAT NEW VALUES MUST BE READ INTO ARRAY HH BEFORE THAT CHUNK IS MULTIPLIED
C BY HH.  THIS SYSTEM MINIMIZES PERIPHERAL PROCESS TIME BY UTILIZING THE
C PERIODICITY OF THE TRANSFORMED GREENS FUNCTION.
C
C
C
C     TWO TOP VIEWS OF LOWER HALF OF EXTENDED MESH(N BY N BY NH/2) - IROW 1
C          AND 2 OF JCOLUMN 1 AND 2 CONSTITUTE THE ACTIVE MESH(N/2 BY N/2 BY
C          NH/2).  THE DIRECTIONS ARE X(I) AND OMEGAX(I) - DOWN ON PAGE.
C          Y(J) AND OMEGAY(J) - TO RIGHT ON PAGE, Z(K) AND OMEGAZ(K) - OUT OF
C          PAGE.
C
C
C                    JCOLUMN                                   JCOLUMN
C                 1   2   3   4                             1   2   3   4
C
C               *****************                         *****************
C               *   *   *   *   *                         * + * + *   *   *
C       IROW=1  * 1 * 5 *   *   *                 IROW=1  * 1 * 3 * 2 * 4 *
C               *****************                         *****************
C               *   *   *   *   *                         * + * + *   *   *
C       IROW=2  * 2 * 6 *   *   *                 IROW=2  * 9 *11 *10 *12 *
C               *****************                         *****************
C               *   *   *   *   *                         * + *   *   *   *
C       IROW=3  * 3 * 7 *   *   *                 IROW=3  * 7 * 5 * 8 * 6 *
C               *****************                         *****************
C               *   *   *   *   *                         * + *   *   *   *
C       IROW=4  * 4 * 8 *   *   *                 IROW=4  *15 *13 *16 *14 *
C               *****************                         *****************
C
C       DISK FILES ON WHICH CHUNKS                ORDER IN WHICH CHUNKS ARE
C       ARE STORED                                MULTIPLIED BY APPROPRIATE
                                                  PORTION OF TRANSFORMED
                                                  GREENS FUNCTION
C
C
```

```
C
      COMMON/ALLCOM/N,NO2,N21,NO4,N34,NH,NHO2,NH21,I2A,I2B,I3A,I3B
      COMMON/TRANCOM/RHO1(32,32,16),RHO2(32,32,16),RHO3(32,32,16),
     1    RHO4(32,32,16),HH(32,32,17)
      COMMON/HN21COM/HN21(65,17)
C THE INITIALIZING OVERLAY OR STAR ADVANCING OVERLAY STORES THE DENSITY
C CHUNKS OF IROW 1 AND 2 FOR JCOLUMN=1 ON DISK FILES 1 AND 2 RESPECTIVELY
C AND FOR JCOLUMN=2 ON DISK FILES 5 AND 6 RESPECTIVELY.  THE GETPHI OVERLAY
C REPLACES THE DENSITY ON THESE DISK FILES WITH THE CORRESPONDING VALUES OF
C POTENTIAL WHICH ARE THEN USED IN THE STAR ADVANCING OVERLAY.  THIS IS
C ACCOMPLISHED THROUGH CALLING SUBROUTINES ANLX(JCOLUMN), ANLSYN(IROW) AND
C SYNX(JCOLUMN) AS DETAILED BELOW.
C
C
C SUBROUTINE ANLX(JCOLUMN) READS RESPECTIVELY IROW 1 AND 2 FROM THE
C FOLLOWING DISK FILES - 1 AND 2 FOR JCOLUMN=1,  - 5 AND 6 FOR JCOLUMN=2.
C IT THEN PERFORMS A PERIODIC ANALYSIS IN THE X DIRECTION OVER JCOLUMN FOR
C I=1,N AND WRITES THE RESULTS RESPECTIVELY FOR IROW 1,2,3, AND 4 ON THE
C FOLLOWING DISK FILES - 1,2,3, AND 4 FOR JCOLUMN=1,  - 5,6,7, AND 8 FOR
C JCOLUMN=2.
      CALL ANLX(1)
      CALL ANLX(2)
C SUBROUTINE ANLSYN(IROW) READS RESPECTIVELY JCOLUMN 1 AND 2 FROM THE
C FOLLOWING DISK FILES - 1 AND 5 FOR IROW=1,  - 2 AND 6 FOR IROW=2,  - 3 AND
C 7 FOR IROW=3,  - 4 AND 8 FOR IROW=4.  IT THEN PERFORMS A PERIODIC ANALYSIS
C IN THE Y DIRECTION OVER IROW FOR J=1,N.  FOR EACH CHUNK IT THEN PERFORMS A
C PERIODIC ANALYSIS IN THE Z DIRECTION FOR K=1,NH, ELEMENT BY ELEMENT
C MULTIPLICATION WITH A SIMILARLY SHAPED CHUNK OF THE TRANSFORMED GREENS
C FUNCTION AND THEN A PERIODIC SYNTHESIS IN THE Z DIRECTION FOR K=1,NH.  THE
C RESULT FOR K=1,NH/2 IS THEN PERIODICALLY SYNTHESIZED IN THE Y DIRECTION
C OVER IROW FOR J=1,N.  THIS LAST RESULT FOR JCOLUMN 1 AND 2 IS WRITTEN
C RESPECTIVELY ON THE FOLLOWING DISK FILES - 1 AND 5 FOR IROW=1,  - 2 AND 6
C FOR IROW=2,  - 3 AND 7 FOR IROW=3,  - 4 AND 8 FOR IROW=4.  THE ORDER IN
C WHICH ANLSYN IS CALLED FOR IROW 1 THROUGH 4 MINIMIZES READING FROM DISK
C FILE 9 OF CHUNKS OF THE TRANSFORMED GREENS FUNCTION AS MENTIONED ABOVE.
      CALL ANLSYN(1)
      CALL ANLSYN(3)
      CALL ANLSYN(2)
      CALL ANLSYN(4)
C SUBROUTINE SYNX(JCOLUMN) READS RESPECTIVELY IROW 1,2,3, AND 4 FROM THE
C FOLLOWING DISK FILES - 1,2,3, AND 4 FOR JCOLUMN=1,  - 5,6,7, AND 8 FOR
C JCOLUMN=2.  IT THEN PERFORMS A PERIODIC SYNTHESIS IN THE X DIRECTION OVER
C JCOLUMN FOR J=1,N.  IT THEN WRITES THE RESULT RESPECTIVELY FOR IROW 1 AND
C 2 ON THE FOLLOWING DISK FILES - 1 AND 2 FOR JCOLUMN=1,  - 5 AND 6 FOR
C JCOLUMN=2.
      CALL SYNX(1)
      CALL SYNX(2)
      RETURN
      END
      SUBROUTINE ANLX(JCOLUMN)
      COMMON/ALLCOM/N,NO2,N21,NO4,N34,NH,NHO2,NH21,I2A,I2B,I3A,I3B
      COMMON/TRANCOM/RHO1(32,32,16),RHO2(32,32,16),RHO3(32,32,16),
     1    RHO4(32,32,16),HH(32,32,17)
      COMMON Z(1025), Y(1025)
      IF(JCOLUMN.EQ.2) GO TO 2
      READ(1) RHO1
      REWIND 1
      READ(2) RHO2
      REWIND 2
      GO TO 3
    2 CONTINUE
      READ(5) RHO1
      REWIND 5
      READ(6) RHO2
      REWIND 6
    3 CONTINUE
      CALL GETSET(3,I2A)
      DO 10 K=1,NHO2
      DO 10 J=1,NO4
      DO  5 I=1,NO4
      Z(I)=RHO1(I,J,K)
      Z(NO4+I)=RHO2(I,J,K)
```

```
      Z(NO2+I)=0.
    5 Z(N34+I)=0.
      CALL FTRANS(3,I2A)
      DO 10 I=1,NO4
      RHO1(I,J,K)=Y(I)
      RHO2(I,J,K)=Y(NO4+I)
      RHO3(I,J,K)=Y(NO2+I)
   10 RHO4(I,J,K)=Y(N34+I)
      IF(JCOLUMN.EQ.2) GO TO 12
      WRITE(1) RHO1
      REWIND 1
      WRITE(2) RHO2
      REWIND 2
      WRITE(3) RHO3
      REWIND 3
      WRITE(4) RHO4
      REWIND 4
      GO TO 15
   12 CONTINUE
      WRITE(5) RHO1
      REWIND 5
      WRITE(6) RHO2
      REWIND 6
      WRITE(7) RHO3
      REWIND 7
      WRITE(8) RHO4
      REWIND 8
   15 RETURN
      END
      SUBROUTINE ANLSYN(IROW)
      COMMON/ALLCOM/N,NO2,N21,NO4,N34,NH,NHO2,NH21,I2A,I2B,I3A,I3B
      COMMON/TRANCOM/RHO1(32,32,16),RHO2(32,32,16),RHO3(32,32,16),
     1     RHO4(32,32,16),HH(32,32,17)
      COMMON/HN21COM/HN21(65,17)
      COMMON Z(1025), Y(1025)
      GO TO(1,2,3,4) IROW
    1 CONTINUE
      READ(1) RHO1
      REWIND 1
      READ(5) RHO2
      REWIND 5
      GO TO 5
    2 CONTINUE
      READ(2) RHO1
      REWIND 2
      READ(6) RHO2
      REWIND 6
      GO TO 5
    3 CONTINUE
      READ(3) RHO1
      REWIND 3
      READ(7) RHO2
      REWIND 7
      GO TO 5
    4 CONTINUE
      READ(4) RHO1
      REWIND 4
      READ(8) RHO2
      REWIND 8
    5 CONTINUE
      CALL GETSET(3,I2A)
      DO 10 K=1,NHO2
      DO 10 I=1,NO4
      DO  7 J=1,NO4
      Z(J)=RHO1(I,J,K)
      Z(NO4+J)=RHO2(I,J,K)
      Z(NO2+J)=0.
    7 Z(N34+J)=0.
      CALL FTRANS(3,I2A)
```

```
   DO 10 J=1,NO4
   RHO1(I,J,K)=Y(J)
   RHO2(I,J,K)=Y(NO4+J)
   RHO3(I,J,K)=Y(NO2+J)
 0 RHO4(I,J,K)=Y(N34+J)
   GO TO(30,49,75,75) IROW
 9 CONTINUE
i0 CONTINUE
   READ(9) HH
i0 JCOLUMN=1
   DO 70 I=1,NO4
   DO 70 J=1,NO4
   DO 52 K=1,NHO2
   Z(K)=RHO1(I,J,K)
i2 Z(NHO2+K)=0.
   CALL GETSET(3,I3A)
   CALL FTRANS(3,I3A)
   IF(IROW.NE.3) GO TO 300
   IF(I.NE.1)GO TO 300
   LL=J
   GO TO 200
i4 DO 70 K=1,NHO2
'0 RHO1(I,J,K)=Y(K)
   GO TO 100
'4 CONTINUE.
   READ(9) HH
'5 JCOLUMN=2
   DO 95 I=1,NO4
   DO 95 J=1,NO4
   DO 77 K=1,NHO2
   Z(K)=RHO2(I,J,K)
'7 Z(NHO2+K)=0.
   CALL GETSET(3,I3A)
   CALL FTRANS(3,I3A)
   IF(IROW.NE.3) GO TO 300
   IF(I.NE.1)GO TO 300
   LL=NO4+J
   GO TO 200
'9 DO 95 K=1,NHO2
'5 RHO2(I,J,K)=Y(K)
   GO TO 125
'0 JCOLUMN=3
   DO 120 I=1,NO4
   DO 120 J=1,NO4
   DO 101 K=1,NHO2
   Z(K)=RHO3(I,J,K)
'1 Z(NHO2+K)=0.
   CALL GETSET(3,I3A)
   CALL FTRANS(3,I3A)
   GO TO(103,105,107,115) IROW
 3 IF(J.NE.1)GO TO 300
   LL=I
   GO TO 200
 5 IF(J.NE.1)GO TO 300
   LL=NO4+I
   GO TO 200
'7 IF(I.NE.1.AND.J.NE.1)GO TO 300
   IF(I.EQ.1.AND.J.EQ.1)GO TO 111
   IF(I.EQ.1)GO TO 109
   LL=I
   GO TO 200
 9 LL=J
   GO TO 200
 1 LL=N21
   GO TO 200
 5 IF(J.NE.1) GO TO 300
   LL=NO4+I
   GO TO 200
 7 DO 120 K=1,NHO2
:0 RHO3(I,J,K)=Y(K)
   GO TO(74,74,400,390) IROW
:5 JCOLUMN=4
```

```
      DO 145 I=1,NO4
      DO 145 J=1,NO4
      DO 127 K=1,NHO2
      Z(K)=RHO4(I,J,K)
  127 Z(NHO2+K)=0.
      CALL GETSET(3,I3A)
      CALL FTRANS(3,I3A)
      IF(IROW.NE.3) GO TO 300
      IF(I.NE.1)GO TO 300
      LL=NO4+J
      GO TO 200
  129 DO 145 K=1,NHO2
  145 RHO4(I,J,K)=Y(K)
      GO TO (400,400,49,49) IROW
  200 DO 205 K=2,NHO2
      Z(K)=Y(K)*HN21(LL,K)
  205 Z(NHO2+K)=Y(NHO2+K)*HN21(LL,K)
      Z(1)=Y(1)*HN21(LL,1)
      Z(NH21)=Y(NH21)*HN21(LL,NH21)
      GO TO 310
  300 DO 305 K=2,NHO2
      Z(K)=Y(K)*HH(I,J,K)
  305 Z(NHO2+K)=Y(NHO2+K)*HH(I,J,K)
      Z(1)=Y(1)*HH(I,J,1)
      Z(NH21)=Y(NH21)*HH(I,J,NH21)
  310 CALL GETSET(4,I3A)
      CALL FTRANS(4,I3A)
      GO TO(54,79,117,129) JCOLUMN
  390 REWIND 9
  400 CALL GETSET(4,I2A)
      DO 410 K=1,NHO2
      DO 410 I=1,NO4
      DO 405 J=1,NO4
      Z(J)=RHO1(I,J,K)
      Z(NO4+J)=RHO2(I,J,K)
      Z(NO2+J)=RHO3(I,J,K)
  405 Z(N34+J)=RHO4(I,J,K)
      CALL FTRANS(4,I2A)
      DO 410 J=1,NO4
      RHO1(I,J,K)=Y(J)
  410 RHO2(I,J,K)=Y(NO4+J)
      GO TO(415,420,425,430) IROW
  415 CONTINUE
      WRITE(1) RHO1
      REWIND 1
      WRITE(5) RHO2
      REWIND 5
      GO TO 435
  420 CONTINUE
      WRITE(2) RHO1
      REWIND 2
      WRITE(6) RHO2
      REWIND 6
      GO TO 435
  425 CONTINUE
      WRITE(3) RHO1
      REWIND 3
      WRITE(7) RHO2
      REWIND 7
      GO TO 435
  430 CONTINUE
      WRITE(4) RHO1
      REWIND 4
      WRITE(8) RHO2
      REWIND 8
  435 RETURN
      END
      SUBROUTINE SYNX(JCOLUMN)
      COMMON/ALLCOM/N,NO2,N21,NO4,N34,NH,NHO2,NH21,I2A,I2B,I3A,I3B
      COMMON/TRANCOM/RHO1(32,32,16),RHO2(32,32,16),RHO3(32,32,16),
     1    RHO4(32,32,16),HH(32,32,17)
      COMMON Z(1025),Y(1025)
      IF(JCOLUMN.EQ.2) GO TO 1
```

```
      READ(1) RHO1
      REWIND 1
      READ(2) RHO2
      REWIND 2
      READ(3) RHO3
      REWIND 3
      READ(4) RHO4
      REWIND 4
      GO TO 2
    1 CONTINUE
      READ(5) RHO1
      REWIND 5
      READ(6) RHO2
      REWIND 6
      READ(7) RHO3
      REWIND 7
      READ(8) RHO4
      REWIND 8
    2 CONTINUE
    4 CALL GETSET(4,I2A)
      DO 10 K=1,NHO2
      DO 10 J=1,NO4
      DO  5 I=1,NO4
      Z(I)=RHO1(I,J,K)
      Z(NO4+I)=RHO2(I,J,K)
      Z(NO2+I)=RHO3(I,J,K)
    5 Z(N34+I)=RHO4(I,J,K)
      CALL FTRANS(4,I2A)
      DO 10 I=1,NO4
      RHO1(I,J,K)=Y(I)
   10 RHO2(I,J,K)=Y(NO4+I)
      IF(JCOLUMN.EQ.2) GO TO 12
      WRITE(1) RHO1
      REWIND 1
      WRITE(2) RHO2
      REWIND 2
      GO TO 15
   12 CONTINUE
      WRITE(5) RHO1
      REWIND 5
      WRITE(6) RHO2
      REWIND 6
   15 RETURN
      END
```

TABLE A3

Array Dimensions
(Program of Table A2)

Array name	General dimensions (note 1)	Dimensions used in actual runs and listing of Table A2	Dimensions used in Figs. A1-A6	Overlays in which dimensioned		
				Star adv. and initl.	GETH	GETPHI
PHI(active)	n x n x h	64 x 64 x 16	8 x 8 x 4	x		
OI	(n/2) x (n/2) x h	32 x 32 x 16	4 x 4 x 4	x		
H	(n+1) x (n+1) x (h+1)	65 x 65 x 17	9 x 9 x 5		x	
HH	(n/2) x (n/2) x (h+1)	32 x 32 x 17	4 x 4 x 5		x	x
HN21 (note 2)	(n+1) x (h+1)	65 x 17	9 x 5		x	x
RHO1, RHO2 RHO3, RHO4	(n/2) x (n/2) x h	32 x 32 x 16	4 x 4 x 4			x
Extended PHI	(2n) x (2n) x (2h)	128 x 128 x 32	16 x 16 x 8	not actually dimensioned (note 3)		

Note 1: The notation a x b x c represents the array dimensions of the subscripts x, y and z, respectively, (or the subscripts ξ, η and ζ, respectively, of the transformed array) such that a x b x c equals the total number of array elements. The Fortran variables N and NH are equal to 2n and 2h, respectively.

Note 2: HN21 is a two-dimensional array containing a boundary plane of $\tilde{H}_{\xi,\eta,\zeta}$ elements. Its first subscript corresponds to ξ or η equivalently, while its second subscript corresponds to ζ.

Note 3: While the program uses smaller arrays in order to avoid dimensioning the (2n) x (2n) x (2h) extended PHI array of Fig. 1, its mathematical existence is necessary for the Fourier solution of the potential of an isolated galaxy.

TABLE A4

Storage of the Fourier Transformed Green's Function $\tilde{H}$
on Desk File 9

(Program of Table A2)

Record No. of file 9	Storage sequence within GETH overlay (Note 1)	Use sequence within GETPHI overlay (Note 2)
1	A	(1,1),(1,3)
2	B	(1,2),(1,4),(3,2),(3,4)
3	A	(3,1),(3,3)
4	C	(2,1),(2,3)
5	D	(2,2),(2,4),(4,2),(4,4)
6	C	(4,1),(4,3)

Note 1: Within the GETH overlay, this is the location in the H array (as designated by letters A-D of Fig. A4) from which "do loop" transfer is made to the HH array followed by writing on the indicated record of disk file 9.

Note 2: Following reading of the indicated record of disk file 9 into the HH array within the GETPHI overlay, this is the sequence of locations in the extended PHI array (as designated by "chunks" (IROW,JCOLUMN) of Fig. A2) upon which z-direction one-dimensional array operations are performed. These operations include multiplication by $\tilde{H}$, the appropriate portion of which is now contained in the HH array. This method minimizes reading of file 9 by using the periodicity and symmetry of $\tilde{H}$.

TABLE A5

Outline of the GETPHI Overlay
(Program of Table A2)

(Refer to Fig. A6 for orientation of arrays RH01, RH02, RH03 and RH04 and to Fig. A2 for file numbers corresponding to the "locations" of these arrays.)

	Listing line Nos. of Table A2
A. CALL ANLX(1): Fig. A6(a).	
1. Read files 1 and 2 into RH01 and RH02, respectively.	299-302
2. Set RH03=RH04=0.	316-317
3. Perform Fourier transform in x-direction over RH01, RH02, RH03 and RH04: $\rho_{x,y,z} \rightarrow \tilde{\rho}_{\xi,y,z}$	310-323
4. Write RH01, RH02, RH03 and RH04 onto files 1, 2, 3 and 4, respectively.	325-332
B. CALL ANLX(2): Fig. A6(b).	
1. Read files 5 and 6 into RH01 and RH02, respectively	305-308
2. Same as steps A.2 and A.3	
3. Write RH01, RH02, RH03 and RH04 onto files 5, 6, 7 and 8, respectively.	335-342
C. CALL ANLSYN(1): Fig. A6(c).	
1. Read files 1 and 5 into RH01 and RH02, respectively	353-356
2. Set RH03=RH04=0	382-383
3. Perform Fourier transform in y-direction over RH01, RH02, RH03 and RH04: $\tilde{\rho}_{x,y,z} \rightarrow \tilde{\rho}_{\xi,\eta,z}$	376-389
4. Read record 1 of file 9 into HH	393
5. For each one-dimensional array in z-direction of which RH01 is composed:	
a. Transfer to one-dimensional array Z, dimensioned at least 2h+1	
b. Set Z=0 for $z \geq h$	
c. Perform Fourier transform in z-direction over Z for $0 \leq z \leq 2h-1$ with the result appearing in one-dimensional array Y: $\tilde{\rho}_{\xi,\eta,z} \rightarrow \tilde{\rho}_{\xi,\eta,\zeta}$	
d. Multiply Y by $\tilde{H}_{\xi,\eta,\zeta}$ to form $\tilde{\phi}_{\xi,\eta,\zeta} = \tilde{\rho}_{\xi,\eta,\zeta} \tilde{H}_{\xi,\eta,\zeta}$	
e. Perform inverse Fourier transform in z-direction over Y and store result for $0 \leq z \leq h-1$ in RH01: $\tilde{\phi}_{\xi,\eta,\zeta} \rightarrow \tilde{\phi}_{\xi,\eta,z}$	
6. Repete step C.5 for RH03	426-454,471-483
7. Read record 2 of file 9 into HH	410
8. Repete step C.5 for RH02 and RH04	411-424,456-469,477-483
9. Perform inverse Fourier transform in y-direction over RH01, RH02, RH03 and RH04: $\tilde{\phi}_{\xi,\eta,z} \rightarrow \tilde{\phi}_{\xi,y,z}$	486-497
10. Write RH01 and RH02 onto files 1 and 5, respectively.	500-503

	Listing Line Nos. of Table A2
D. CALL ANLSYN(3): Fig. A6(e).	
1. Read files 3 and 7 into RH01 and RH02, respectively.	365-368
2. Same as steps C.2-C.9 except for sequencing of reading tape 9 into HH and the z-directional operations. Table A4 details this sequencing.	
3. Write RH01 and RH02 onto files 3 and 7, respectively.	512-515
E. CALL ANLSYN(2): Fig. A6(d).	
1. Same as step D except that files 2 and 6 correspond to RH01 and RH02, respectively, for read and write operations.	
F. CALL ANLSYN(4): Fig. A6(f).	
1. Same as step D except that files 4 and 8 correspond to RH01 and RH02, respectively, for read and write operations.	
G. CALL SYNX(1): Fig. A6(a).	
1. Read files 1, 2, 3 and 4 into RH01, RH02, RH03 and RH04, respectively.	530-537
2. Perform inverse Fourier transform in x-direction over RH01, RH02, RH03 and RH04: $\mathfrak{F}_{\xi,y,z} \to \phi_{x,y,z}$	550-560
3. Write RH01 and RH02 onto files 1 and 2, respectively.	562-565
H. CALL SYNX(2): Fig. A6(b).	
1. Read files 5, 6, 7 and 8 into RH01, RH02, RH03 and RH04, respectively.	540-547
2. Same as step G.2	
3. Write RH01 and RH02 onto files 5 and 6, respectively.	568-571

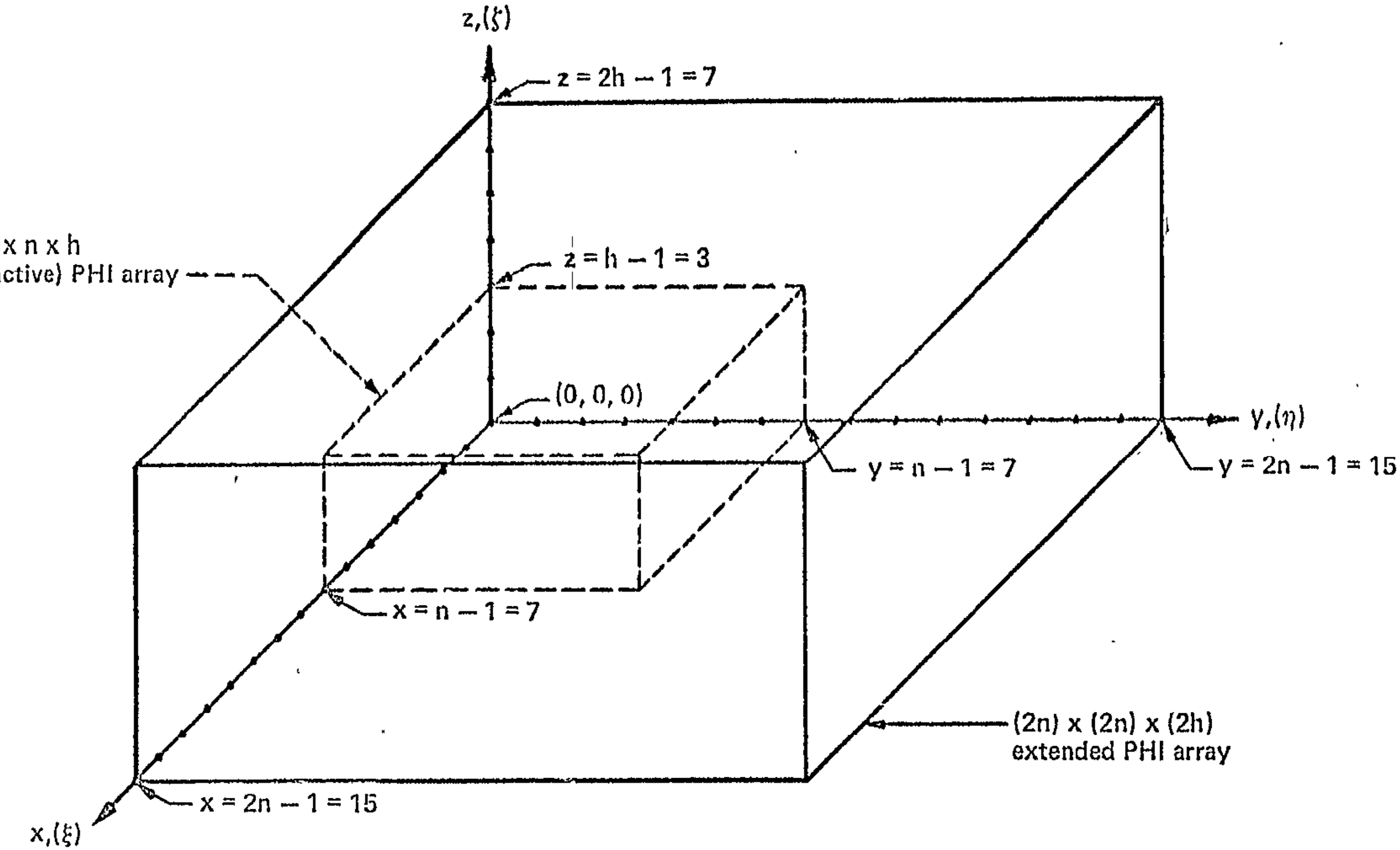

Figure A1.- PHI array (active), which contains the galactic density/potential mesh, and the extended PHI array, which is required for the Fourier potential solution of an isolated galaxy. Each x-, y-, or z-axis represents the following: (a) the x-, y-, or z-spacial direction; (b) the untransformed array subscript x, y, or z; and (c) the x-, y-, or z-direction transformed array subscript ξ, η or ζ, respectively. For clarity in this and the following figures, the PHI array is dimensioned n x n x h = 8 x 8 x 4 while in the program as listed in Table A2 and as actually run it is dimensioned 64 x 64 x 16 (Table A3 refers).

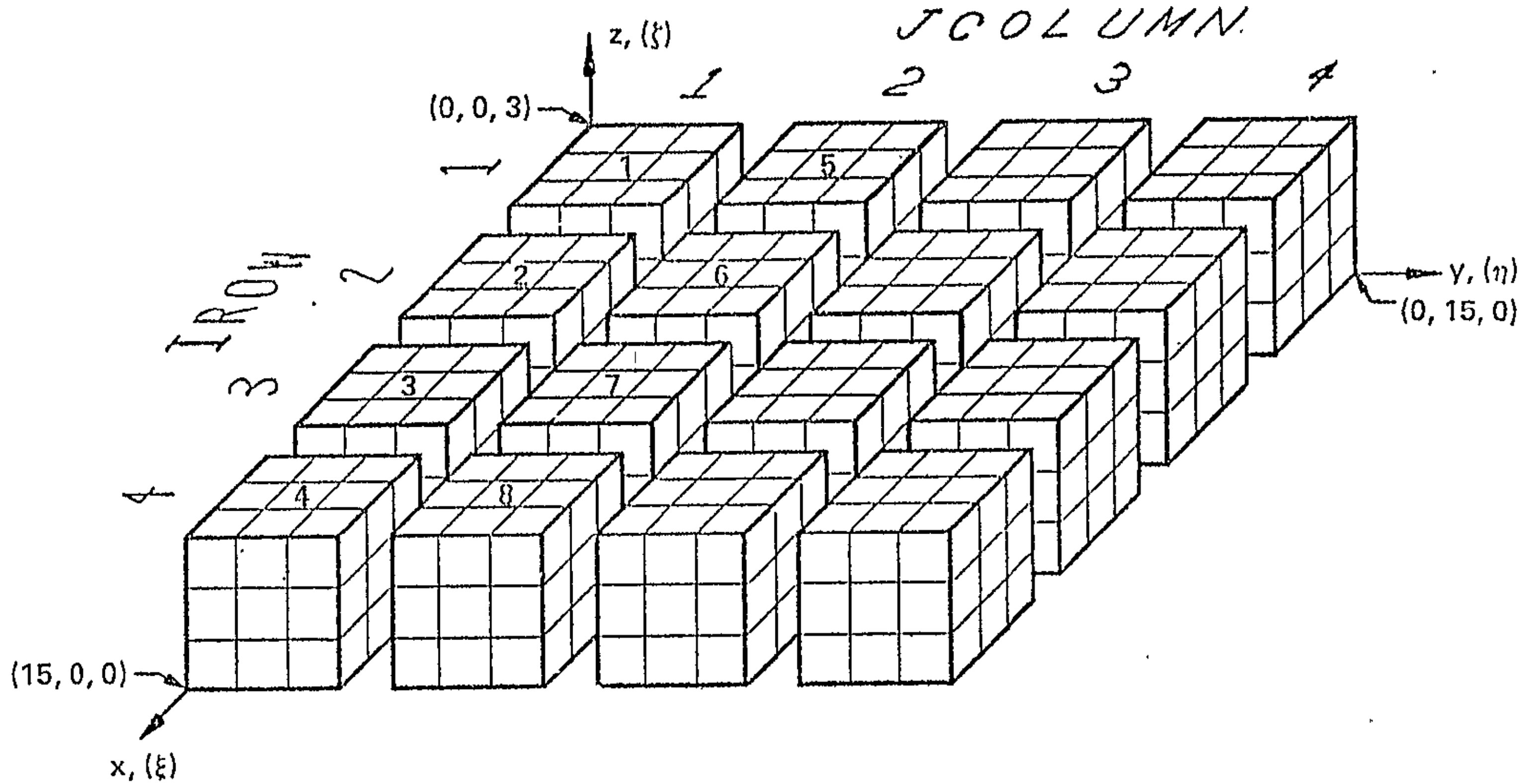

Figure A2.- (Program of Table A2) - Lower half ($0 \leq z \leq h-1 = 3$) of the extended PHI array showing row and column designations of "chunks." IROW 1 and 2 of JCOLUMN 1 and 2 constitute the active PHI array. The numbers on "chunks" of JCOLUMN 1 and 2 indicate the numbers of the disk files on which those chunks are stored. The "chunks" of JCOLUMN 3 and 4 do not require disk file storage.

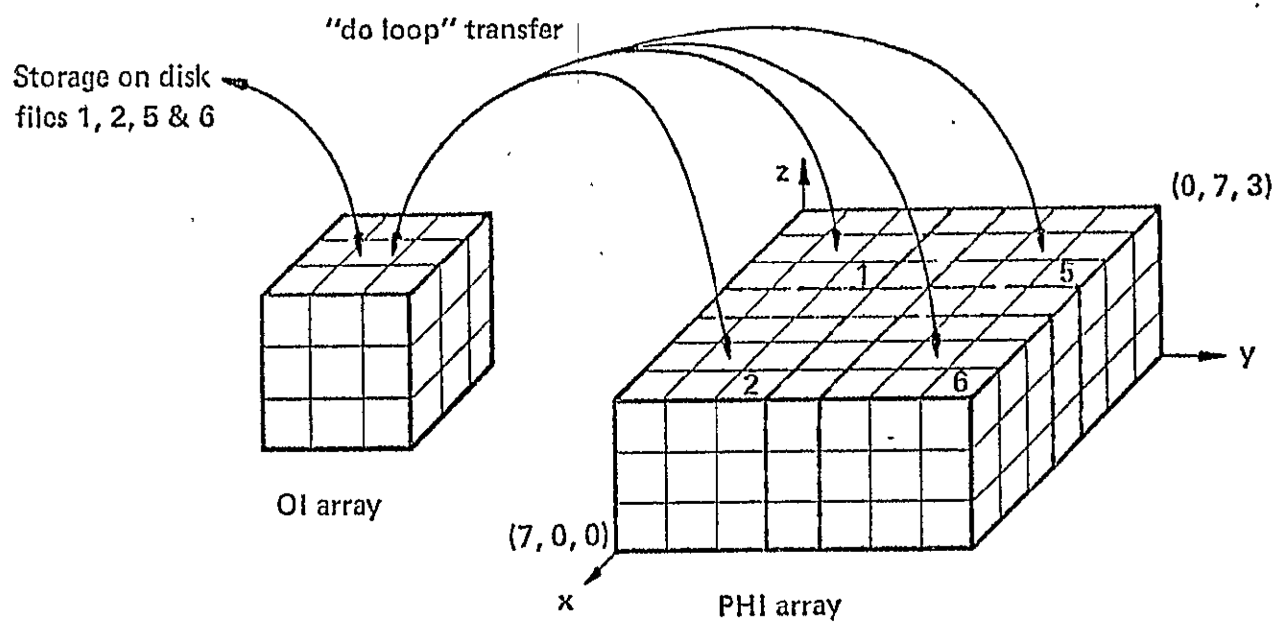

Figure A3.- (Program of Table A2) - Arrays dimensioned in the initializing and star advancing overlays. The numbers on the "chunks" indicate the disk files on which they are stored.

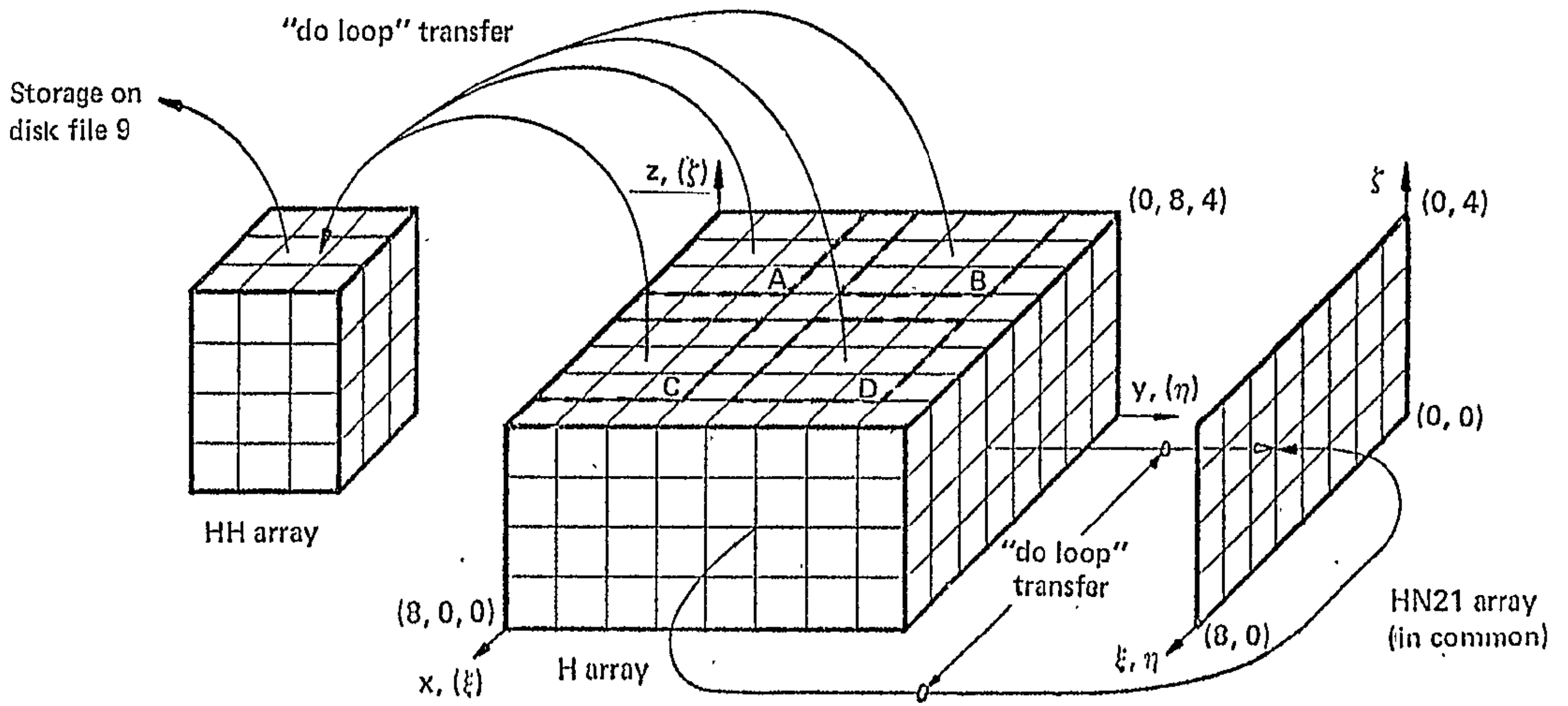

Figure A4.- (Program of Table A2) - Arrays dimensioned in GETH overlay, which performs a Fourier transform of the Greens function $H_{x,y,z}$ and stores the resulting $\tilde{H}_{\xi,\eta,\zeta}$. (Letters A, B, C and D are referenced by Table A4.)

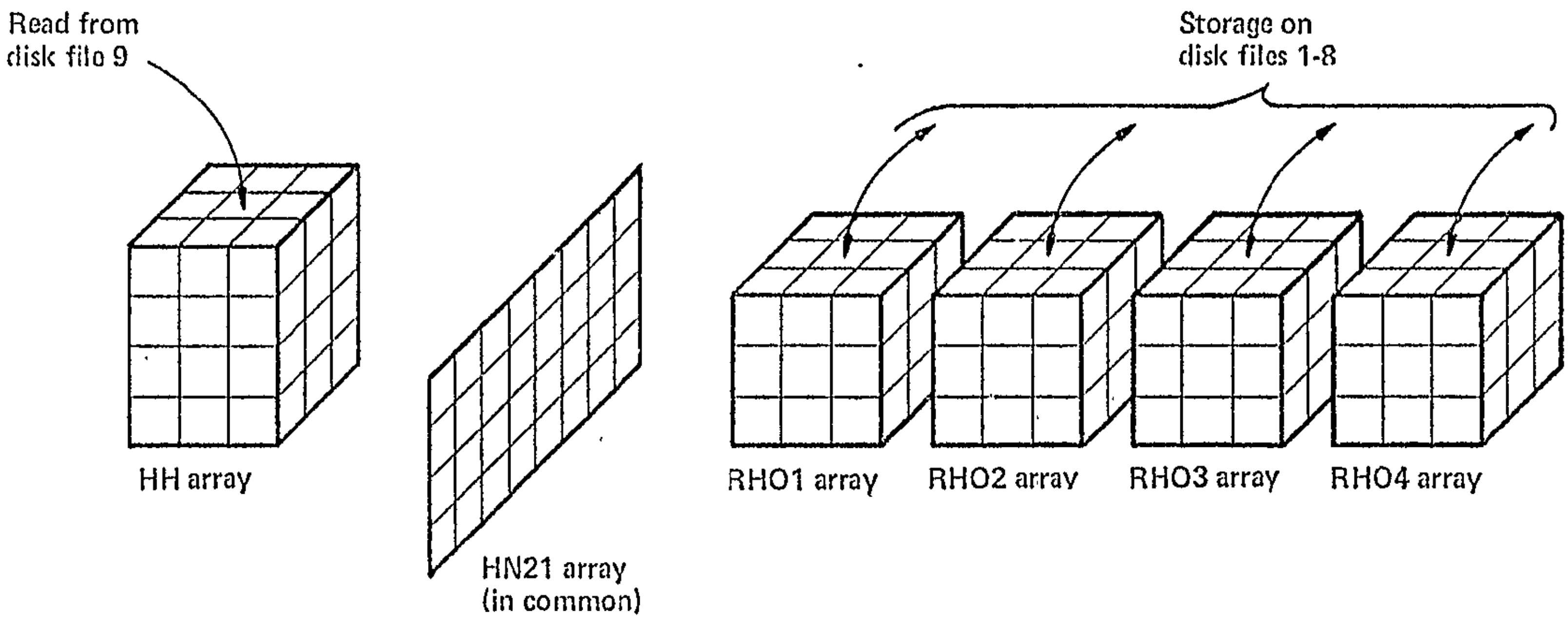

Figure A5.- (Program of Table A2) - Arrays dimensioned in the GETPHI overlay, which solves for the potential of an isolated galaxy.

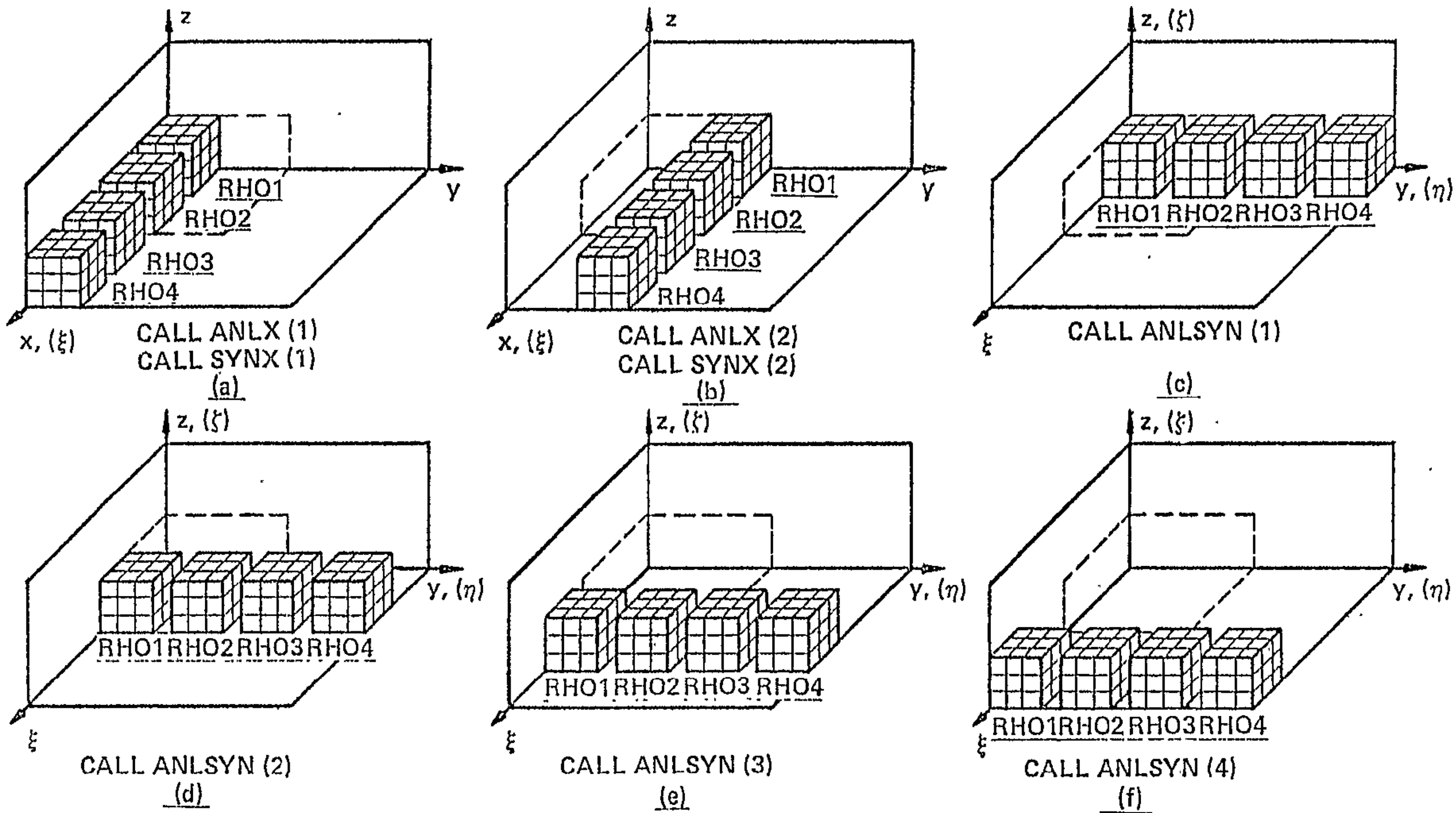

Figure A6.- (Program of Table A2) - Alignment of arrays RHO1, RHO2, RHO3, and RHO4 during calls by the GETPHI overlay to its subroutines. Although the active PHI array and the extended PHI array are not dimensioned within the GETPHI overlay, their projections on the planes $x = 0$, $y = 0$, and $z = 0$ are represented by dashed and solid lines, respectively. Axes labels represent subscripts of array elements which are untransformed (x,y,z), transformed (ξ,η,ζ) or either, as appropriate.