An Inverse-Problem Approach to Cluster Dynamics
Chakrabarty, D.; Portegies Zwart, S.F.

Published in:
Astronomical Journal

DOI:
10.1086/423215

Citation for published version (APA):

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: http://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.

UvA-DARE is a service provided by the library of the University of Amsterdam (http://dare.uva.nl)
AN INVERSE-PROBLEM APPROACH TO CLUSTER DYNAMICS

DALIA CHAKRABARTY
Department of Physics and Astronomy, Rutgers University, 136 Frelinghuysen Road, Piscataway, NJ 08854-8019

AND

SIMON PORTEGIES ZWART
Astronomical Institute Anton Pannekoek and Section Computational Science, University of Amsterdam, Kruislaan 403, NL-1098 SJ Amsterdam, Netherlands

Received 2003 September 15; accepted 2004 June 9

ABSTRACT

We propose a new nonparametric algorithm that can be implemented to study and characterize stellar clusters. The scheme attempts to simultaneously recover the stellar distribution function and the cluster potential by using projected radii and velocity information about the cluster members. The pair of these functions that is most consistent with the input data is detected by the Metropolis algorithm. In this work, the cluster characteristics recovered by the algorithm (“CHASSIS”) are calibrated against N-body realizations of two clusters, the Hyades and the Arches. The cluster mass and line-of-sight projected velocity dispersion profiles are correctly reproduced by the algorithm when the cluster obeys the assumption used in the code, namely, isotropy in phase space. The results recovered by the code are shown to be insensitive to the choice of the initial parameters. The results are also not influenced by increasing the number of input data points as long as this number exceeds a minimum value, which is moderately low for an input data set that obeys the assumptions of isotropy and sphericity. 

Key words: Galaxy: center — Galaxy: kinematics and dynamics

1. INTRODUCTION

The characterization of stellar clusters is, more often than not, an exercise in model building. One usually begins with an assumption (of sphericity) for the cluster geometry. Quite often, luminosity measurements are used in conjunction with fitting algorithms to provide the surface brightness profile of the cluster (Larsen et al. 2001). However, a major shortcoming of such a procedure is that such fits are often unstable over the whole radial range considered. Moreover, the robustness of these fitting techniques is always challenged by details in the observed data. It is possible in principle to obtain smooth approximations to the surface density profile and projected velocity dispersion profile from observations. These projected quantities could be inverted to yield their total or three-dimensional counterparts via the Abel integral equation; such a deprojection is unique in the spherical case. Unfortunately, an inversion of this nature is highly sensitive to shot noise in the data bins. Genzel et al. (1996) acknowledge this problem and adopt the more secure alternative of starting with parameterized forms of the density and radial velocity dispersion profiles, which are projected back to observational space, while maintaining the search for those projected profiles that fit the observed result best. Such an inverse approach, whenever possible, is almost always the better option, since it involves less error. Frequently, a cluster is considered to be in virial equilibrium; velocity dispersion estimates or enclosed-mass information is then extracted from the virial equation (Bosch et al. 2001; Ghez et al. 1998). A method that can substitute this assumption of a virialized cluster is certainly more satisfactory, since observations alone cannot be sufficient to establish the validity of this conjecture.

This state of affairs can be improved upon by considering the inverse-problem approach. It is indeed true that such a formalism demands a moderately large data set. Kinematic information about the members of a number of observed stellar clusters suffices to fulfill this requirement. In this paper, we propose a nonparametric technique that can be used for the characterization of stellar systems. The algorithm presented here has been used before to estimate the mass enclosed in the inner few parsecs of the Milky Way (Chakrabarty & Saha 2001, hereafter Paper I). We refer to our scheme as CHASSIS (characterization of stellar systems using a new inverse scheme).

This paper aims to advance the case of using CHASSIS in investigations of stellar clusters. In order to establish the resourcefulness of this code, we test our inverse algorithm on two simulated star clusters, which were placed at known distances and projected on a selected part of phase space. We present the results of analyzing these simulated star clusters with CHASSIS and then compare the results with the cluster simulations. This comparative exercise reflects the degree of robustness of the algorithm.

2. INITIAL CONDITIONS AND CALCULATIONS

As input for the simulations, we adopt two clusters that we use as templates for the initial conditions. We call these models A (mimicking the Arches star cluster) and H (mimicking the Hyades star cluster; see Table 1). The cluster simulations were performed in three steps: (1) initialization of the stellar system, (2) calculation of the evolution of the cluster to a certain time, and (3) preparation of the simulated star cluster to use as input for the inverse algorithm. Details about the Hyades model and the global setup of the calculations for the Arches model have already been published (Portegies Zwart et al. 2001, 2002). One ingredient of the Arches model that makes it different compared with previously performed N-body calculations is the inclusion of the external tidal field of the Galaxy and the frictional drag force on the bound cluster. As a result, the cluster spirals in from an initial distance of 6 pc from the Galactic center to less
than 2 pc, in about a million years. The strong tidal field in model A causes the cluster to become strongly aspherical and elongated perpendicular to the line connecting the cluster with the Galactic center. However, CHASSIS works on the assumption of sphericity of the cluster, as explained below. In light of this obvious contradiction, the results obtained from the inverse algorithm for the Arches cluster were expected to be interesting. Model A was run on a GRAPE-6 and took about 188 CPU-hours. Model H took about 55 hours on a single-board GRAPE-4.

For the selected models, positions and velocities of all stars are calculated self-consistently together with the evolution of the stars and binaries in the clusters. The calculations are performed using the Starlab software environment and utilize the GRAPE-4 and GRAPE-6 special-purpose computers (Makino et al. 1997). The simulations were stopped and ported to the inverse algorithm when the age of the cluster was about 1 Myr for model A and about 400 Myr for model H. For model H, all the stars remaining in the cluster after 400 Myr were ported to the inverse algorithm, while for model A the stars remaining after 1 Myr were used. From these sets, various smaller data sets were selected for direct input into CHASSIS (see below).

### 2.1. Preparation of the Clusters for the Inverse Algorithm

The stellar samples used as inputs in our algorithm are selected from the N-body data sets that describe the two clusters. We use kinematic data for a maximum of about 2600 stars belonging to the Hyades in our work. The number of stars in the Arches from which the input data are chosen is much higher (about 55,000). The input files for the algorithm are prepared by choosing stars either at a random or by picking stars according to their luminosity. The kinematic information provided in the N-body data sets about each star is made to undergo appropriate transformations in order to calculate the corresponding apparent position, radial velocity, and transverse velocity. These quantities are the inputs for CHASSIS. Diagrammatic representations of the kinematics of the two clusters are given in Figures 1 and 2. These are plots of radial and transverse velocities against radial positions of the stars on the plane of the sky, that is, the apparent positions of the stars.

The N-body simulations indicated that neither are the clusters isotropic nor are they perfectly spherical. The extent of the presence of anisotropy and asphericity in the A and H models is brought out in Figure 3, which displays the radial

![Figure 1](image1.png)

**Fig. 1.**—Plots of radial (left) and transverse (right) velocities of stars in model H against apparent positions. The data are obtained from N-body calculations described in § 2. To construct this plot, we have chosen every third star in the N-body representation of this cluster.

<table>
<thead>
<tr>
<th>Model</th>
<th>$W_0$</th>
<th>$N$</th>
<th>$M$ ($M_\odot$)</th>
<th>$R_{hm}$ (pc)</th>
<th>$R_t$ (pc)</th>
<th>$T$ (Myr)</th>
<th>$N$</th>
<th>$M$ ($M_\odot$)</th>
<th>$R_{hm}$ (pc)</th>
<th>$R_{gc}$ (pc)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>3</td>
<td>64K</td>
<td>62600</td>
<td>0.17</td>
<td>0.52</td>
<td>6.0</td>
<td>0.97</td>
<td>55224</td>
<td>54200</td>
<td>0.093</td>
</tr>
<tr>
<td>H</td>
<td>6</td>
<td>3K</td>
<td>1606</td>
<td>2.50</td>
<td>13.7</td>
<td>13K</td>
<td>400</td>
<td>2628</td>
<td>1192</td>
<td>4.66</td>
</tr>
</tbody>
</table>

### Notes.

The second through seventh columns give the initial density profile (King [1966] model parameter), the number of stars, the total cluster mass, the virial radius, and the tidal radius of the cluster and its distance to the Galactic center. The subsequent columns give the parameters for these models at the moment the data were analyzed by the inverse algorithm: the time, the number of remaining stars bound to the cluster, the total mass of these stars, the half-mass radius, and the distance to the Galactic center.

---

1 See http://manybody.org.
dependence of the ratios of the x- and y-components of the velocity dispersion vector ($\sigma_x$ and $\sigma_y$) to the z-component of the same ($\sigma_z$) and the ratios of the x- and y-components of the spatial dispersion vector ($\sigma_z$ and $\sigma_x$) to the z-component of the same ($\sigma_z$). Such ratios are only crude estimates of the deviation from sphericity and isotropy, but they serve our purpose of estimating the degree to which we can expect the input data to be incongruous with the assumptions of the code.

As can be seen from this figure, in model H the ratios of the spatial dispersions are about 1 to a radius of about 5 pc, after which both $\sigma_z/\sigma_x$ and $\sigma_y/\sigma_z$ rise slowly with radius, though a marked discontinuity in the run of the spatial dispersions shows up just inside a radius of about 10 pc. A big jump occurs in the ratios $\sigma_x/\sigma_z$ and $\sigma_y/\sigma_z$ between about 4 and 7 pc. This picture prompts us to choose to work within about 10 pc in the Hyades cluster, knowing that the kinematic distribution of these input data is not very well consistent with isotropy in this radial range.

The equivalent radial cutoff for model A appears to be about 0.2 pc, although farther on the cluster becomes highly aspherical and anisotropic. So we choose to select kinematic information from model A from radii within 0.2 pc; however, when viewed at a finer resolution, even within this radius, the run of the dispersion ratios is quite jerky, indicating that the data from the Arches cluster do not abide by the assumption of isotropy in the code.

Fig. 2.—Same as Fig. 1, but for model A. This N-body cluster has 55,224 stars in it, but only stars at pre-fixed intervals in projected radius have been included in the plot.

Fig. 3.—Ratio of the x- and y-components of the velocity (and spatial) dispersion, $\sigma_x$ and $\sigma_y$ (and $\sigma_z$ and $\sigma_x$), to the z-component of the velocity (and spatial) dispersion, $\sigma_z$ (and $\sigma_z$), for models H (left) and A (right).
The chosen radial ranges are also inclusive of the half-mass radii of the two clusters, about 4.66 pc for the Hyades and 0.093 pc for the Arches. From models H and A, the masses enclosed within 4.6 and 0.093 pc are approximately 584 and 24,522 $M_\odot$, respectively. These values are very close to half the mass of the cluster (see Table 1).

3. THE ALGORITHM

The code used in our work was introduced and discussed in detail in Paper I. Observed kinematic data (apparent positions of stars on the plane of the sky and their velocities) are used as input to this algorithm. CHASSIS is a nonparametric code that attempts to identify both the equilibrium stellar distribution function (DF) from which the observed kinematic data are drawn and the potential of the observed stellar system. The code assumes sphericity of this stellar system. In its present form, the scheme also assumes an isotropic DF. Paper I reports a test that was developed to check the consistency of the data used with the assumption of isotropy. The basic design of our algorithm is described briefly below.

As an initial guess, the code requires a set of functions in the form of an arbitrarily chosen trial potential $\Phi$ (rather the corresponding density profile $\rho$) and distribution function $f$, which is a function of energy $E$ only, via the assumption of isotropy. The final results have no bearing on the initial configurations. In order to obtain a potential from a chosen density, a discretized density profile is used; that is, $\rho(r)$ is held constant over the radial bin defined around radius $r$. The calculation of the potential of a series of spherical mass shells is simple and is given in Paper I. The only constraints placed on these trial functions is that the density profile be monotonically decreasing in radius and the DF be monotonically increasing in effective energy. Adherence to these conditions is demanded every time the profiles are altered to generate new profiles. Such tweaking of the functions takes place at the end of each iterative step.

At every iteration, the current DF is projected into observable space, using the observed kinematic data. Thus, if we have information on the line-of-sight (LOS) velocity $v_p$ and the apparent position $r_p$, then

$$v_p(r_p, v_p) = \int dz \int dv_x dv_y f(v_x^2 + v_y^2 + 2\Phi(r)).$$

(1)

Here $r$ is the spherical radius and $r_p$ the cylindrical radius on the plane of the sky, so that $r^2 = x^2 + y^2 = r^2 - z^2$; $v_x$ and $v_y$ are components of the velocity parallel and tangential to the radius $r$. We choose the $y$-axis to lie completely in the plane containing the radius vector $r$ and the LOS. This implies that $v_r^2 = (v_x \sin \theta)^2 + (v_y \cos \theta)^2$, $\theta$ being the polar angle, that is, $\cos \theta = z/r$.

This projection of $f(E)$ to $\nu_p(r_p, v_p)$ could be achieved if a smooth approximation to $f$ were considered. Another even simpler way (introduced in Merritt 1996), used in our work, is to hold $f$ constant over any integral cell defined around a given value of $E$. Then, corresponding to this energy bin, the projected distribution function is

$$\nu_p^{\text{cell}}(r_p, v_p) = \int dz A(r, r_p, v_p),$$

(2)

where $A(r, r_p, v_p) = \int dv_x dv_y$ is the area that the energy bin occupies in phase space. Estimation of this area requires knowledge of the bounds on the individual phase-space coordinates that correspond to each energy bin. Identification of these bounds is discussed in Paper I. The total projected distribution function is a simple sum over all the energy bins:

$$\nu_p(r_p, v_p) = \sum_i f_i(\nu_p^{\text{cell}}(r_p, v_p)).$$

(3)

The product of all the projected distribution functions, each obtained for a pair of $r_p$ and $v_p$ in the observed data set, gives a likelihood function $L$:

$$\log L = \sum_{i=1}^N \nu_{p,i},$$

(4)

where $N$ is the total number of pairs of data points in the set and $\nu_{p,i}$ corresponds to the $i$th data point.

When the accessible velocity data include proper motion $v_p$ (where $v_p = (v_x^2 + v_y^2)^{1/2}$) rather than LOS velocity data, the projection equation analogous to equation (1) is

$$\nu_{p, i}(r_p, v_p) = \int dz \int f(E) dv_z,$$

(5)

while if the data include complete velocity information, the projection is simplest and is given by

$$\nu_{p, \text{3D}}(r_p, v_p, v_p) = \int dz f(E).$$

(6)

The likelihood function is therefore a function in $n$ dimensions, where $n = n_r \times n_E$. Here $n_r$ is the number of radial bins corresponding to the chosen density profile and $n_E$ is the number of energy bins characterizing the DF. The global maximum of the highly nonlinear function $L$ is sought in this $n$-dimensional space by the Metropolis algorithm. A striking feature of using this method of optimization is that we are ensured a whole array or ensemble of models distributed according to the likelihood, rather than the single model associated with the maximum likelihood. The availability of the ensemble allows us to estimate uncertainties simply by measuring the spread of values across the ensemble. This is a big advantage of using the Metropolis algorithm. The maximization scheme is elaborated upon in Paper I.

Thus, in CHASSIS, at the $i$th iterative step, the DF and density profile are slightly tweaked from their forms used during the $(i - 1)$th step. The choice of these functions for the $i$th step is referred to as $x_i$. The current $f$ is projected into observable space, at the currently chosen potential for each data point, and the likelihood for this step $L(x_i)$ is calculated. At the beginning, Metropolis is expected to wander around in the multidimensional domain of the likelihood function, and $L$ tends to increase during this period. Soon, Metropolis settles down to an “equilibrium” phase during which the likelihood is likely to be close to the optimal value. Convergence is characterized in our code by the step $k$ at which $L(x_k) < L(x_{k-N_{\text{iter}}})$. Here $N_{\text{iter}} + 1$ is the total number of iteration steps that we use in any run. We record $x_{k-N_{\text{iter}}}, \ldots, x_k$ as our likelihood-weighted sample of $\rho$ and $f$.

From the recorded DF and density profiles, the mass within a pre-fixed radius $r$ is calculated as follows:

$$M(r) = \int_0^r \rho(r) 4\pi r^2 dr.$$  

(7)
In discretized form, this equation is represented as

$$M(r) = \frac{4\pi}{3} \left[ \delta^3 \sum_{i=0}^{j} \rho_i (3i^2 - 3i + 1) + \rho_j (r^3 - j^3) \right], \quad (8)$$

where $r$ lies in the range $[j\delta, (j+1)\delta]$, that is, $j = \text{int}(r/\delta)$. The quantity $\delta$ is the width of a radial bin, and $j \in \mathbb{N}$ such that $j \leq \text{int}(r_{\text{max}}/\delta)$.

The velocity dispersion profile $\sigma(r)$ is calculated from the moments of $f(E)$:

$$\sigma(r)^2 = \frac{\int f(E) v^2 d^3v}{\int f(E) d^3v} \cdot \left( \frac{\int f(E) v d^3v}{\int f(E) d^3v} \right)^2, \quad (9)$$

where $v^2 = v_r^2 + v_\phi^2$ and $E = v^2/2 + \Phi(r)$. Thus, we obtain a value of $\sigma(r_p)$ for every $r_p$ in the input data set. This equation can be molded into the following difference equation:

$$\sigma(r)^2 = 2 \left[ \frac{3}{5} \sum_{i=0}^{j} \delta_i^3 P_1 - 3f_j (\Phi(r)^{3/2} - \delta_i^{3/2}) \right]$$

$$- \left[ \frac{3}{4} \sum_{i=0}^{j} \delta_i^2 P_2 - 4f_j (\Phi(r)^{3/2} - \delta_i^{3/2}) \right]^2, \quad (10)$$

where

$$\gamma_i = \frac{\delta_i}{i} - \frac{\Phi}{r^2} \quad (11)$$

and

$$P_1 = 5i^4 - 10i^3 + 10i^2 - 5i + 1,$$

$$P_2 = 3i^2 - 3i + 1,$$

$$P_3 = 4i^3 - 6i^2 + 4i^2 - 1.$$
and distribution function as the initial guess in the algorithm. The initial density function is given by

\[ \rho(r) = \rho_0 \left( 1 + \frac{r^2}{3r_c^2} \right)^\alpha, \]  

where \( \rho_0 \) is maintained constant at 5000 and different choices of \( r_c \) and \( \alpha \) determine different initial density profiles. Changing \( \rho_0 \) was found to have no bearing on the results, while the results appeared most sensitive to the choice of \( \alpha \). The effect of changing the initial density profiles was found to diminish with an increased number of iterative steps, \( N_{\text{iter}} \). At \( N_{\text{iter}} = 4000 \), density profiles defined by \( \alpha \) lying in the range \([-0.3, -3.5]\) and \( r_c \in [1, 7.5] \) were found to imply similar results.

The initial DF has a normalization of unity and is chosen to have a simple power-law form,

\[ f(\epsilon) = (C + \epsilon)^\beta, \]

where \( \epsilon \) is the effective energy. A number of trial runs with different values of the constant \( C \) showed that the results are independent of it even with a relatively low \( N_{\text{iter}} \). Thus, we used \( C = 0 \) in all our main runs. The parameter \( \beta \), on the other hand, had a greater effect on the results. We used initial DFs

![Fig. 5.—Profiles of the Hyades cluster, as estimated by CHASSIS, from runs performed with various initial configurations. The left panels display the enclosed-mass profiles, while the velocity dispersion profiles recovered from the different runs are shown in the right panels. In the top panels, the index \( \beta \) is held constant (at 5.5) and \( \alpha \) is changed. The profile for \( \alpha = -1.5 \) is marked by solid error bars with superposed filled circles at the 50% level; those for \( \alpha = -2.5 \) are in dotted lines with open circles, and error bars for \( \alpha = -3.5 \) are in dashed lines with crosses. In the bottom panels, the initial density profile is maintained the same (\( \alpha = -1.5 \)) but the initial choice of DF varies in steepness: \( \beta = 5.5, 6.5, 7.5 \) are marked by solid, dotted, and dashed error bars, superposed with filled circles, open circles, and crosses, respectively. The results obtained from the calculations for the data describing model H are shown by solid black curves. The error bars are at the 16% and the 84% levels. In the dispersion profiles, the error bar at any radius is calculated as the difference between the dispersion value corresponding to the value of density at the 84% level and at the 16% level. The input for the algorithm is the set of the 37 brightest stars in model H.](image)
characterized by values $\beta \in [1, 10.5]$. Any effect of the steepness of the initial choice of the DF on the results could be negated by using a large enough $N_{\text{data}}$ (4000 was found to suffice, in the sense that the results were found to be consistent within the error bars).

5. RESULTS FROM MODEL H

The cluster properties that CHASSIS directly provides are the density and the equilibrium DF of the stellar sample selected from the $N$-body cluster. These cluster parameters are shown for the Hyades in Figure 4. Since these data include information on all three components of the velocity of each star, we use both LOS velocity and proper motion in our algorithm. Thus, the appropriate projection equation is given by equation (6). The results obtained for this data set are discussed below.

5.1. Independence from Initial Model

In this section, we present evidence to establish the insensitivity of the results to the initial guess used in the algorithm. This is demonstrated by means of the concurrence of the enclosed-mass and velocity dispersion profiles of the Hyades cluster recovered by runs done with different initial guesses. For all these runs, the input to the code consists of 37 stars, selected by their luminosity, from model H.

From Figure 5, one can see that changing the initial configuration results in profiles that agree with each other within the error bars. This provides us with confidence about the robustness of the code insofar as changing the initial guess is concerned.

5.2. Number of Data Points

We conducted a series of numerical experiments to study the effect that the number of data points used as inputs to the algorithm has on the results suggested by the code. Samples of different sizes were selected, by their luminosity, from the data set corresponding to the $N$-body calculations that describe the Hyades cluster. The mass and velocity dispersion profiles of the cluster were calculated for these different input files and were compared with each other, as well as with their $N$-body counterparts. One quantity that is of interest is the mass within the half-mass radius of the cluster. This is, of course, derived from the enclosed-mass profile, but being a simple scalar quantity, it can aid in gauging the extent to which the size of the input file is effective in determining results. The effect is shown in Figure 6. In addition to the value of the enclosed mass from different runs, the figure also depicts how well the predicted mass estimates tally with the $N$-body value. The cluster profiles are shown in Figure 7.

Figures 6 and 7 show that the number of stars for which kinematic information is fed into the algorithm is not particularly influential in determining the mass profile accurately. It is however true in general that the errors in the calculation of the cluster parameters that are directly recovered by the code are reduced with an increase in the input file size. This is discussed in § 7.

5.3. Selection of Stars

A series of runs were undertaken to appraise the influence of the mode of selection of stars from the $N$-body realization of the cluster. The kinematic information for these chosen stars then forms the input file for the inverse algorithm. One way to pick the stars from the $N$-body data set is to choose a certain number of stars at random. Alternatively, we could sort the stars by their luminosity and accept the $N$ brightest stars to form an input data file of size $N$ (here $N$ is any integer). We seek to explore the effect of these different methods of building up the input data for CHASSIS.

In Figure 7, the enclosed-mass and velocity dispersion profiles of the Hyades as predicted by the code are compared with the known ($N$-body) cluster configuration. The functions are obtained from runs done with input data files of diverse sizes, formed by choosing the constituent stars either randomly or by luminosity, from model H. The results show that when the stars are chosen by their luminosity, the match between the estimated and $N$-body quantities is in general better than when the choice is at random. In fact, Figure 7 indicates that CHASSIS does a good job in reproducing the cluster profiles when the stars are chosen by luminosity. The results from the run done with the larger number of stars (199 stars) is particularly good. Even the run done with a much smaller number of stars (17) offers results that are consistent with the $N$-body mass profiles, within the error bars, which are indeed rather larger. However, the overlap between the velocity dispersion profile obtained from the latter run is not so good. One reason for this general lack of agreement between the estimated velocity dispersion and the $N$-body dispersion profile is a shortcoming in the very method of extraction of the dispersion profile from the $N$-body data set. This is explained in detail in § 8.

6. RESULTS FROM MODEL A

$N$-body simulations of the Arches cluster gave us our model A. We have performed a number of runs with data selected from this model and fed into CHASSIS, in order to obtain the characteristics of the Arches cluster. Runs were implemented to evaluate the effects of changing the size of the input data set and of changing the manner of selecting stars from model A. The $N$-body simulation of this cluster was
performed at much lower radii compared with that for the Hyades. One interesting difference between the two clusters is that the Arches cluster is identified as highly aspherical in the \( N \)-body calculations. Isotropy in velocity space is also a worse assumption for model A than for model H. We have confined ourselves to the very central core of the Arches (radius \( < 0.15 \) pc), where the effects of asymmetry in phase space are not too prominent (Fig. 3).

Figure 8 shows the mass within the central 0.093 pc (the virial radius, approximately) estimated by the code for input data sets of various sizes. This predicted value is compared with the \( N \)-body value of 24,522 \( M_\odot \) for the enclosed mass.

The effect of the process of selecting the stars from the \( N \)-body realization of the Arches cluster is displayed in Figure 9. The figure shows the cluster profiles obtained from a number of runs that were performed with different-sized input files constructed by picking stars randomly from model A. From these runs, it is clear that the inverse algorithm overestimated the enclosed mass at larger radii. When the stars are chosen in decreasing order of their luminosity, the runs show a better match with the \( N \)-body mass profiles, within the error bars. The compatibility with the known velocity dispersion profiles was likewise poorer for these runs.

7. DISCUSSION

The aim of this paper was to advance the case for using CHASSIS to characterize stellar clusters. We proceeded in this exercise by first introducing the salient features of the inverse
Similarly promising results were obtained from runs done with within the error bars (as well as with the \(N\) recovered mass profiles agreed with each other very well /\(C_0\) 3.5 at a fixed choice of the initial distribution function. The of the initial density profile varying from \(11\) the initial guess for the cluster characteristics (see Fig. 5). Results were compared from runs done with the steepness \(C_0\) the initial file size.

Fig. 8.—Same as Fig. 6, but for the mass enclosed within the inner 0.093 pc of the Arches cluster (half-mass radius). The mass estimates are from runs done with input files of various sizes, constructed by picking stars by luminosity from model A. The \(N\)-body estimate of the mass within this radius is about 24,522 \(M_\odot\) (solid line).

good, though when seen over short length scales the degree of overlap is not as good as in the case of the mass profiles. This is in fact a feature that is consistently borne out by all the presented velocity dispersion profiles.

The signatures of the clusters, as known from the \(N\)-body simulations, were compared with the characteristics recovered by the code. The degree of overlap between the simulated cluster configurations and the predicted ones is expected to shed light on the applicability of the algorithm. The relevant cluster attributes were judged to be the enclosed-mass and line-of-sight projected velocity dispersion profiles.

We tested the robustness of the algorithm to variations in the initial guess for the cluster characteristics (see Fig. 5). Results were compared from runs done with the steepness \(C_0\) of the initial density profile varying from \(-1.5\) to \(-2.5\) and \(-3.5\) at a fixed choice of the initial distribution function. The recovered mass profiles agreed with each other very well within the error bars (as well as with the \(N\)-body profile). Similarly promising results were obtained from runs done with a fixed initial density profile and a DF that varied in steepness (\(\beta\) was changed from \(5.5\) to \(6.5\) and \(7.5\)). In regard to the velocity dispersion profiles, it is seen that the overall match is good, though when seen over short length scales the degree of overlap is not as good as in the case of the mass profiles. This is in fact a feature that is consistently borne out by all the presented velocity dispersion profiles.

This discrepancy owes more to the way the known dispersion is drawn out from the \(N\)-body simulations of the clusters than it does to the calculation of the dispersion by CHASSIS. The projected radii from the \(N\)-body data sets are binned and the velocities of the stars in each such bin are calculated. (For model H, stars with velocities lower than 1 km s\(^{-1}\) were used in this calculation, while for the Arches, stars in each bin with velocities lower than 20 km s\(^{-1}\) were used.) This discrete distribution of velocity dispersions is plotted in Figure 10 as crosses. This distribution is “optimally” smoothed to repre-

sent the velocity dispersion of the cluster. By “optimal” smoothing is meant smoothing done (using the SuperMongo software) with a filter size that, if exceeded, renders the plot less jagged but does not alter the overall shape. This smoothed run of the dispersion with \(r_p\) is what the calculated (by CHASSIS) dispersion profile is compared with. This method of extracting the dispersion profile from the \(N\)-body results is indeed ad hoc to some extent, but the methodology is adopted consistently; consequently, for the purposes of comparison, it is good enough.

There is another technical reason why in general the match between the calculated and known velocity dispersion profiles is less good than that between the enclosed-mass profiles. The mass profile of the relevant cluster is estimated from its density by means of equation (8), while the velocity dispersion is calculated using equation (9). Since the latter equation involves two summations, as opposed to the single one in equation (8), it is to be expected that any error in the estimate of the DF will imply a greater deviation in the evaluated dispersion value than any error in the estimated density will contribute to the mass profile.

The quantities that the inverse algorithm directly offers are the density profile and the equilibrium distribution function of the set of stars for which kinematic data are used as input to the code. These quantities are then used to calculate the mass and dispersion profiles, under the assumptions of sphericity and isotropy in velocity space. These assumptions were essentially made to ease our calculations but are not necessarily realistic constraints on clusters. For example, we recognize the Arches cluster as highly aspherical from the \(N\)-body simulations. In order to be able to predict stable results with CHASSIS, we confined our analysis to radii within which the degree of asphericity and anisotropy is markedly low.

The way to ensure that the set of stars used as input to CHASSIS is closer to a spherical configuration is by using the kinematic data of the more massive stars. These stars form a subset in the cluster that is relatively more spherical. It is after all the tidal field of the Galaxy that, when included in the \(N\)-body simulations, causes the cluster to elongate in one direction. This field has a weaker effect on the heavier than the less massive stars in the cluster, since the more massive stars tend to reside closer to the center of the cluster owing to mass segregation. This is confirmed by Figure 11. The left panel in this figure represents the run of the ratio of the transverse to the radial velocity components of 200 stars chosen from model A. In one case the stars were chosen by luminosity (open circles), while in the other case they were chosen randomly (filled triangles). The distribution of the ratio \(v_t/v_r\) with \(r_p\) is more centrally concentrated when the more massive (i.e., roughly speaking, the brighter) stars are included in the sample—there are comparatively more triangles at higher radii. The right panel of the figure shows that the effect of the tidal field of the Galaxy is more pronounced in the sample constructed by random selection of the stars. The value of the apparent position is indexed by a number in the data set representing the sample; the ratio \(r_p/z\) is plotted against this number for the two samples. The distributions of the ratios are smoothed by the same amount in both cases and show that the sample of stars constructed by random choice is less spherical than when the choice is by mass.

In this context, let us reiterate that the choice on the basis of stellar mass roughly resembles the choice based on luminosity. In fact, a few runs were carried out with the \(N\) most massive stars, and the profiles obtained from these runs corroborated the results already mentioned.
Thus, if the input set of \( N \) stars (\( N \) a natural number) is formed by choosing the \( N \) most massive stars within a certain radius instead of choosing \( N \) stars randomly, the assumption of sphericity is more closely obeyed and the enclosed-mass profile as recovered by CHASSIS is expected to be more compatible with the \( N \)-body profile. In fact, this is true even interior to the radial cutoffs that we have imposed in our selection of the stars from the \( N \)-body outputs (of 10 and 0.15 pc for the Hyades and Arches, respectively), as verified by Figure 11.

If the set of stars used as input to the algorithm form an aspherical configuration, then it is to be expected that our code will overestimate the mass (see Fig. 12). This is what is seen from runs done with input data sets that are formed by choosing stars randomly and is evident in the bottom panels of Figures 7 and 9.

We have also explored the effect of changing the number of data points (\( N \)) that are used as input to the code. Naively, one may conclude that a larger number of data points invariably improves the result, in the sense that the error bars diminish in size. The larger the data set, the narrower the region around the maximum of the likelihood function in the multidimensional energy-radius space. Consequently, the wandering of the Metropolis algorithm in its “equilibrium” phase is smaller, thus bringing down the extent of the errors. It is also expected that increasing the number of input data points will soon lead to nearly stable results.

We noted that until the number of stars in the input data set reaches about 80, the simulated and predicted quantities are closer for the Hyades cluster than for the Arches (see Figs. 7 and 9). One reason for this is that the Arches cluster is known from the \( N \)-body simulations to be much more aspherical than the Hyades. The degree of anisotropy in velocity space is also greater in the Arches. As stated in § 2, even in the central 0.15 pc, the ratios of the different components of the velocity...
(and spatial) dispersion, when sought as functions of radius, undulate about a mean of unity. The same quantities for the Hyades are less nonlinear inside the chosen radial cutoff for model H (about 10 pc). Thus, for the Arches cluster, stability in the results is attained at a higher value of \( N \) (at about \( N = 80 \)) than when input data are chosen from model H, for which this number is 17. This is evident in the comparison of Figures 6 and 8.

Our code is rather powerful in the sense that it can spot the answers even when \( N \) is quite low. We propose that this potency is a direct outcome of implementing the Metropolis algorithm as the optimizer. Metropolis is a highly competent algorithm and is more effective than seeking the maximum in the likelihood function by brute force, as has been done in other nonparametric projects that used an isotropic distribution function (Merritt & Saha 1993; Merritt & Tremblay 1993). When the likelihood is highly nonlinear, such a method is not strong enough to recover the maximal region of the likelihood unless the number of input data points is sufficiently high to adequately constrain the likelihood. Even when Metropolis is
used, we find that for a model such as model H, we need to have around 100 data points to constrain the enclosed mass at the half-mass radius within a range of about 120 $M_{\odot}$, that is, fractional errors of about 20%.

The approximation of isotropy, as used in the algorithm, aids in simplifying the calculations but needs to be improved by means of the inclusion of a two-integral distribution function in the code. This is planned as a future exercise.

8. CONCLUSIONS

We have carried out an analysis of two $N$-body clusters using the inverse algorithm CHASSIS. Our code aims to simultaneously recover the potential and the distribution function of the stars whose kinematic details are fed in as input. The global maximum in the likelihood function is sought by the Metropolis algorithm. The cluster profiles directly offered by the algorithm are used to calculate the line-of-sight projected velocity dispersion and enclosed-mass profiles of the cluster.

We looked into the robustness of the code with respect to change in the initial parameters used. The nearly concurrent results indicate that the code does indeed withstand variations in the input configurations very well (see Fig. 5).

When input data were used from the model that mimics the Hyades cluster, the enclosed-mass and velocity dispersion profiles predicted by the code were consistent with the simulated profiles for the Hyades cluster as long as the number of kinematic data points used as input was at least 17, though the error bars that accompany the results from this run are very large. For the Arches cluster, an input data set with at least as many as 63 stars is required to yield a mass profile that is consistent with the $N$-body profile within the error bars, though the velocity dispersion profile from this run falls short of the predicted $N$-body dispersions (Fig. 9). The input data set needs to contain kinematic information from at least 80 stars to result in both mass and dispersion profiles that are compatible with the known profile, within the error bars.

The overlap between the quantities suggested by the code and the $N$-body profiles is better when the input model follows the assumption of phase-space isotropy closely, that is, within about 10 pc for the Hyades and 0.15 pc for the Arches cluster.

When the input file comprises the brightest stars, the predictions of the code match the cluster profiles better than when the stars are picked randomly. Roughly speaking, the brightest stars tend to follow the trend of the most massive stars and are segregated toward the cluster center, while the dimmer stars are more prominent in the cluster halo. Thus, an observer who is likely to look toward the center stands a greater chance of picking out those stars for which the code predicts the correct results. Consequently, CHASSIS can claim to be a propitious tool for the analysis of data on stellar clusters.

Increasing the number of stars in the input file does not affect the results beyond decreasing the size of the error bars; for a given model, there is a minimum value of this number that, when exceeded, implies stable results. This value is low when the cluster is very nearly spherical and isotropic. For the Arches cluster this numerical value is around 80, while it is as low as about 17 for the Hyades, though the run done with these 17 stars yields very large error bars; at the half-mass radius of the $N$-body model for the Hyades cluster, the mass enclosed varies in a range from about 500 to 900 $M_{\odot}$. (As $N$ increases, the error bars reduce in size considerably, so that when $N \sim 200$ the mass within the half-mass radius ranges from about 582 to about 586 $M_{\odot}$.) We believe that is it our sophisticated optimization routine, which utilizes the Metropolis algorithm, that helps us to identify the correct answers even with quite small input data sets. The search for the range of $N$ that corresponds to consistent results can be fully automated.

Our numerical experiments offer confidence in the applicability of CHASSIS as a tool to analyze stellar clusters. The algorithm has been shown to be successful when the cluster is spherical and isotropic in velocity space. The assumption of isotropy is an ingredient in the code that is under review.

CHASSIS has been used to estimate the mass of the central black hole in the Galaxy previously (Paper I) and, in this paper, to characterize open stellar clusters. In its present form, it is put forward as a useful tool to understand kinematic data from other stellar systems such as clusters.

D. C. would like to acknowledge P. Saha, who is the joint author of CHASSIS. S. P. Z. was supported by the Royal Dutch Academy of Sciences (KNAW) and by the Netherlands Organization for Scientific Research (NWO). We would also like to thank the anonymous referee, whose careful scrutiny helped to improve the paper considerably.

REFERENCES

King, I. R. 1966, AJ, 71 64