STOCHASTIC DYNAMICS OF GLOBULAR CLUSTERS

JOHN ALEXANDER HAWKIN

Stochastic Dynamics of Globular Clusters

by

© John Alexander Hawkin
Bachelor of Science (Honours) in Computer Science and Physics

A thesis submitted to the
School of Graduate Studies
in partial fulfillment of the
requirements for the degree of
Master of Science.

Department of Computational Science
Memorial University of Newfoundland

December 18, 2006

ST. JOHN'S NEWFOUNDLAND

Your file Votre référence
ISBN: 978-0-494-31252-0
Our file Notre référence
ISBN: 978-0-494-31252-0

Contents

Abstract

The most common approach to studying the dynamics of globular clusters is to calculate and analyze the position and velocity for each star. The rapid variation of these variables is a limitation of this approach. It is perhaps easier to study star cluster dynamics by analyzing more slowyly changing variables, such as the period of a star. In this thesis, I study two particular slow-changing variables of stars in $N-$body simulations, these being the aforementioned theoretical period, and the angle between the star's closest approach to and its farthest retreat from the center of the cluster. Our work involves fitting these variables to ARMA models, both through careful individual analysis and through an automated procedure. While the ARMA models which we considered could not be successfully fit to these variables, it is possible that an analysis using higher order ARMA models, or possibly GARCHMA models, would be more successful.

The second portion of this thesis deals with the distribution of forces in a star cluster. A simple approximation of this distribution was given by Holtsmark in 1917. This approximation assumes that the cluster has an infinite radius and a constant density, and thus it assumes the force distribution is not spatially dependent. We showed from studies of simulations that these assumptions are not valid for a real cluster, because stars on the edge of the cluster do not experience the same force, on average, as stars in the middle. A new force distribution which takes this fact into

account must be used instead.

It would seem that research such as the statistical time series models mentioned above, as well as the new force distribution, could eventually lead to the derivation a new set of dynamical equations for star clusters. This work is not covered in this thesis, but is an obvious and quite likely very fruitful continuation which we hope to explore in the future.

Acknowledgments

First of all I would like to acknowledge my supervisor, Dr. John Lewis, for all of the hard work and time he put into helping me with this thesis. I especially thank him for his work on the derivation of a generalized force distribution for a star cluster, some of which has been presented in this thesis. I learned a great deal from him throughout my time at Memorial, and he was always willing to give up time to help me with my research or my writing, even when he was extremely busy with other commitments. I can only hope his other students will realize how lucky they are to have such a knowledgeable, friendly supervisor.

I would also like to thank my family and friends for helping me out along the way, especially for the last few months. Writing a thesis is always a very time-consuming endeavor, and I appreciate their willingness to work their schedules around mine, and to help me in any other way that they could. Having dedicated friends has helped me a great deal during my most busy times.

I need to give a special thanks to my long time girlfriend, Jenn Doyle. Her support for me has always been consistent and unwavering. I know that this would have been much more difficult for me if I hadn't had the chance to vent my frustrations upon her whenever I needed to. I love her dearly and hope that I can be as supporting of her as she has been of me.

I would like to thank Fred Perry, Chris Stevenson and Mark Staveley for the

computer support they provided. All of the staff in the Physics department also deserve my thanks, and I'm sure there are others that I'm not thinking of, so to everyone who helped me in the last couple of years, thank you.

Last but not least I would like to thank my officemates, Terry and Jason, for putting up with me for so long without ever complaining, at least not while I was around! They have been very supportive of me, and I wish them both the best.

-John Hawkin

Chapter 1

Introduction

1.1 Globular clusters

Globular clusters are groupings of 30,000 to 3,000,000 stars which are tightly packed together, in a close-to-spherical formations. They are found both within our galaxy and in other galaxies. They are of great interest in astrophysics, because they provide insight into many different astrophysical processes and systems. Above all, they are old: most are composed of very old population II stars. They are so old that they can provide a useful lower limit on the age of galaxies (in particular the Milky Way) and for that matter, the universe. Systems of globular clusters in other galaxies can provide valuable information about the dynamics and chemical composition of those galaxies in their early stages of development, and the mere existence of globular clusters within galaxies is probably significant in regard to the origins of these galaxies.

Other than their great age, another advantage of globular clusters is that the integrated properties of the globular clusters in galaxies hundreds of Mpc away can be determined, while in contrast it is very difficult to study individual stars at distances of just a few Mpc. Thus understanding the properties of globular clusters allows collection of data on very distant galaxies.

Though they are almost spherical in shape, some globular clusters are more eccentric than others. Cluster M19 is one of the least spherical globular clusters in our galaxy, with a minor-to-major axis ratio of $\frac{b}{a} \simeq 0.73$. This is extreme: an analysis of 99 globular clusters in our galaxy shows that they have a mean axis ratio of $\frac{b}{a} \simeq 0.93$. This elongation of the ellipsoidal shape of globular clusters is likely due to their rotation [16].

The ages of globular clusters are of great interest, and though uncertainty in the calculation of their ages is relatively high, there is considerably less uncertainty in the calculated age difference between various globular clusters [16]. There are many methods of age determination for globular clusters, a couple of examples being the use of main sequence termination magnitudes and red giant star luminosities [26]. Some of these calculations have revealed that the most metal-rich clusters are several billion years younger than the most metal-poor. Determination of the differences in ages of globular clusters within our own galaxy can reveal important information about this galaxy's formation. It has also been shown that differences in the ages of globular clusters is one of the most important factors in determining their properties [16].

Perhaps the biggest problem associated with globular clusters is what is commonly known as the second parameter problem. The metallicity and age of globular clusters seem to determine most of their properties, but these two factors alone cannot effectively determine all of their properties. Thus, there seems to be some other parameter of importance, in addition to these two. There are many ideas as to what this parameter may be (such as variations in elemental abundances), but this is still a topic of debate [16]. It is worth noting that this is called the second parameter problem simply because originally metallicity was the only parameter known to be of importance, and after the term was coined it was shown that variations in cluster ages are also of importance [16].

1.2 Modeling of globular clusters

1.2.1 Kinetic equations

A kinetic equation is usually defined as a single equation for the phase space density $f(\mathbf{x}, \mathbf{v}, t)$ [37]. Kinetic equations are always of the form

$$\frac{\partial f}{\partial t} = \Gamma(f; \boldsymbol{x}, \boldsymbol{v}, t) \tag{1.1}$$

where Γ is a functional of its arguments [37]. The phase space density is usually introduced in the form that we have given it here, as a function of $\mathbf{x}$, $\mathbf{v}$ and t. It can, however, be expressed as a function of any canonical variables [17].

Now we present the physical interpretation of f, and for this purpose we will use the form $f(\mathbf{x}, \mathbf{v}, t)$, as its physical interpretation is the most intuitive. If we consider a cube centered on $\mathbf{x}$, having side lengths dx, dy, dz, and velocities in the x, y and z directions which fall between v_x and $v_x + dx$, v_y and $v_y + dy$ and v_z and $v_z + dz$, the average number of particles in this cube is given by

$$f(\mathbf{x}, \mathbf{v}, t) d^3\mathbf{x} d^3\mathbf{v} \tag{1.2}$$

where $d^3\mathbf{x} = dxdydz$ and $d^3\mathbf{v} = dv_x dv_y dv_z$. Thus the phase space density defines the one particle density at each point in the phase space, which is described by the canonical coordinate system being used.

Kinetic equations for globular clusters

In this section we introduce two kinetic equations currently used in modeling globular clusters: the collisionless Boltzmann equation and the Fokker-Plank equation.

The collisionless Boltzmann equation is given by [17]

$$\frac{\partial f}{\partial t} + \sum_i a_i \frac{\partial f}{\partial v_i} + \sum_i v_i \frac{\partial f}{\partial x_i} = 0. \tag{1.3}$$

Binney and Tremaine [17] describe this as the fundamental equation of stellar dynamics. It describes a system in which the dynamics of the system are influenced much more by the density distribution than by the movements of individual elements of the system [17]. For a derivation of the collisionless Boltzmann equation, see appendix A.

The Fokker-Plank equation is a more complex kinetic equation in which some effect due to close encounters is considered. It has the form [43]

$$\frac{\partial f}{\partial t} + \sum_i a_i \frac{\partial f}{\partial v_i} + \sum_i v_i \frac{\partial f}{\partial x_i} = -\sum_{i=1}^{3} \frac{\partial}{\partial v_i} \left(f \left\langle \Delta v_i \right\rangle \right) + \frac{1}{2} \sum_{i,j=1}^{3} \frac{\partial^2}{\partial v_i \partial v_j} \left(f \left\langle \Delta v_i \Delta v_j \right\rangle \right) \tag{1.4}$$

where the right hand side of the equation is the collisional term. A derivation of this equation can be found in [43].

1.2.2 Forms of the distribution function

While an appropriate kinetic equation outlines the behavior of the distribution function, in order to obtain a useful model of globular clusters we need to define the form of the distribution function. In this section we introduce some different distribution functions that have been used in modeling globular clusters.

Polytropes and Plummer's model

We will start with a quick review of polytropes. The polytropic model was first used in astronomy not to model star clusters, but as a model governing the structure of individual stars. This was a model in which conduction and radiative transport were considered to be negligible, thus giving the pressure-density relationship of a classical ideal gas:

$$p = K\rho^{\gamma} \tag{1.5}$$

where K is a constant and γ is the ratio of specific heats at constant pressure and volume. This can be put into the more convenient form

$$p = K\rho^{1+\frac{1}{n}} \tag{1.6}$$

where n is known as the polytropic index. For a self-gravitating sphere of polytropic gas the equation of hydrostatic support is given by [17]

$$\frac{dp(r)}{dr} = -\rho(r)\frac{d\Phi(r)}{dr}. \tag{1.7}$$

Substituting eq.(1.6) for p, we have

$$\left(1+\frac{1}{n}\right)K\rho^{\frac{1}{n}-1}\frac{d\rho}{dr} = -\frac{d\Phi}{dr}. \tag{1.8}$$

A couple of useful concepts to introduce at this point are that of relative energy ε and relative potential Ψ. The relative energy ε is given by

$$\varepsilon = -E + \Phi_0 \tag{1.9}$$

where Φ_0 is a reference potential, which is chosen in a convenient way. The relative potential Ψ has the form:

$$\Psi = -\Phi + \Phi_0. \tag{1.10}$$

Now we can rewrite eq.(1.8) in terms of the relative potential Ψ:

$$\left(1+\frac{1}{n}\right)K\rho^{\frac{1}{n}-1}\frac{d\rho}{dr} = \frac{d\Psi}{dr}. \tag{1.11}$$

We use the flexibility of the reference potential and choose Ψ such that $\Psi = 0$ at the boundary of our system. This lets us rearrange and integrate eq.(1.11) to obtain an equation for the density in terms of the potential:

$$\rho^{\frac{1}{n}} = \frac{1}{n} \frac{\Psi}{K\left(1+\frac{1}{n}\right)}. \tag{1.12}$$

At the time of Plummer, polytropes were reasonably well understood models [1]. Thus, it was only natural that an attempt would be made to find a distribution function that would lead to a polytropic model. A convenient such function is given by

$$f(\varepsilon) = \begin{cases} F\varepsilon^{n-\frac{3}{2}}, & (\varepsilon > 0) \\ 0, & (\varepsilon \leq 0) \end{cases} \tag{1.13}$$

where F is a constant. We will now show that this function leads to a polytropic model. From this distribution function we can obtain an equation for density by using the formula

$$\rho(r) = \int f(\varepsilon) d^3\mathbf{v}. \tag{1.14}$$

If the relative energy ε is chosen such that $f(\varepsilon) = 0$ for $\varepsilon \leq 0$, then this becomes [1]

$$\rho(r) = 4\pi \int_0^{\sqrt{2\Psi}} f(\varepsilon)\, v^2 dv \tag{1.15}$$

Using this equation, we find that for the distribution function eq.(1.13), the density is given by

$$\begin{aligned} \rho(r) &= 4\pi \int_0^{\sqrt{2\Psi}} f(\varepsilon)\, v^2 dv \\ &= 4\pi F \int_0^{\sqrt{2\Psi}} \varepsilon^{n-\frac{3}{2}} v^2 dv. \end{aligned} \tag{1.16}$$

We note at this point that we can express the distribution function as

$$\begin{aligned} f &= f(\varepsilon) = f(-E + \Phi_0) = f(-\Phi - \frac{1}{2}v^2 + \Phi_0) \\ &= f(\Psi - \frac{1}{2}v^2). \end{aligned} \tag{1.17}$$

Using this expression in eq.(1.16) we get

$$\rho(r) = 4\pi F \int_0^{\sqrt{2\Psi}} \left(\Psi - \frac{1}{2}v^2\right)^{n-\frac{3}{2}} v^2 dv \tag{1.18}$$

If we make the substitution $v^2 = 2\Psi\cos^2\gamma$, this equation becomes

$$\begin{aligned} \rho(r) &= 2^{\frac{7}{2}}\pi F\Psi^n \int_0^{\frac{\pi}{2}} \sin^{2n-3}\gamma\cos^2\gamma\sin\gamma d\gamma \\ &= c_n\Psi^n \end{aligned} \tag{1.19}$$

where the constant c_n is given by

$$\begin{aligned} c_n &= 2^{\frac{7}{2}}\pi F \int_0^{\frac{\pi}{2}} \sin^{2n-2}\gamma\cos^2\gamma d\gamma \\ &= 2^{\frac{7}{2}}\pi F \left\{ \int_0^{\frac{\pi}{2}} \sin^{2n-2}\gamma d\gamma - \int_0^{\frac{\pi}{2}} \sin^{2n}\gamma d\gamma \right\} \\ &= \frac{(2\pi)^{\frac{3}{2}}\left(n-\frac{3}{2}\right)!F}{n!}. \end{aligned} \tag{1.20}$$

For c_n to be finite, we must have $n > \frac{1}{2}$. Here we note that eq.(1.19) is of the same form as eq.(1.12), with c_n replaced by $\left(\frac{1}{n}\frac{1}{K\left(1+\frac{1}{n}\right)}\right)^n$. Thus we see that eq.(1.13) does indeed lead to a polytropic model.

Recalling Poisson's equation in spherical coordinates

$$\frac{1}{r^2}\frac{d}{dr}\left(r^2\frac{d\Phi}{dr}\right) = 4\pi G\rho \tag{1.21}$$

and using eq.(1.19) to substitute $c_n\Psi^n$ for ρ then

$$\frac{1}{r^2}\frac{d}{dr}\left(r^2\frac{d\Psi}{dr}\right) + 4\pi Gc_n\Psi^n = 0 \tag{1.22}$$

where the relative potential Ψ has been used rather than Φ. Next we substitute the scaled variables $s = \frac{r}{b}$ and $\Psi = \frac{\Psi}{\Psi_0}$ for r and Ψ, and we have [23]

$$\frac{1}{s^2}\frac{d}{ds}\left(s^2\frac{d\Psi}{ds}\right) = \begin{cases} -\Psi^n & \Psi > 0 \\ 0 & \Psi \leq 0 \end{cases} \tag{1.23}$$

This equation is known as the Lane-Emden equation, and we will make use of it shortly.

It is at this point that we introduce Plummer's model, given by the potential function

$$\Psi = \frac{1}{\sqrt{1+\frac{1}{3}s^2}}. \tag{1.24}$$

This potential function describes a polytropic system with an index of 5 ($n = 5$). To see that this is the case, we show that it satisfies the Lane-Emden equation derived above (eq.(1.23)). We get

$$\begin{aligned} \frac{1}{s^2}\frac{d}{ds}\left(s^2\frac{d\Psi}{ds}\right) &= -\frac{1}{3s^2}\frac{d}{ds}\left(\frac{s^3}{\left(1+\frac{1}{3}s^2\right)^{\frac{3}{2}}}\right) \\ &= -\frac{1}{\left(1+\frac{1}{3}s^2\right)^{\frac{5}{2}}} = -\Psi^5. \end{aligned} \tag{1.25}$$

Thus the Lane-Emden equation is satisfied, and eq.(1.24) describes a polytropic system.

From eq.(1.19) we can get the corresponding density for this potential:

$$\rho = c_5\Psi^5 = \frac{c_5\Psi_0^5}{\left(1+\frac{1}{3}s^2\right)^{\frac{5}{2}}}. \tag{1.26}$$

This equation is known as Plummer's Law, because Plummer showed that the density distribution of this model is a reasonable fit for observations of some globular clusters. Though it is somewhat successful at modeling spherical galaxies and globular clusters (which are approximately spherical), this model cannot be successfully applied to elliptical galaxies. This is because the density of elliptical galaxies falls off less steeply than r^{-4}, while as we can see from eq.(1.26), the density for Plummer's model obeys $\rho \sim r^{-5}$ [1].

Isothermal Models

Though Plummer's law presented in the preceding section gives a reasonable fit for the observed density distributions of some globular clusters, there is definitely a need for better models. It turns out that the best models currently used are adaptations of isothermal models, and are known as lowered isothermal models. Before discussing these we will introduce isothermal models.

If the polytropic index of eq.(1.6) is taken to be infinite, the equation becomes

$$p = K\rho. \tag{1.27}$$

This corresponds to the relationship between pressure and density of a classical ideal gas at constant temperature. Thus, models of this form are known as isothermal models. Unfortunately the Lane-Emden equation, eq.(1.23), is no longer well defined at $n = \infty$. So instead we will use the equation of hydrostatic support of a self-gravitating isothermal ball of gas [1], given by

$$\frac{dp}{dr} = -\frac{GM(r)}{r^2}\rho. \tag{1.28}$$

Differentiating eq.(1.27) we obtain

$$\begin{aligned} \frac{dp}{dr} &= K\frac{d\rho}{dr} \\ &= \frac{k_B T}{m}\frac{d\rho}{dr} \end{aligned} \tag{1.29}$$

where $K = \frac{k_B T}{m}$, k_B being Boltzmann's constant and m being the mean mass per particle. Equating eq.(1.28) and eq.(1.29) we have

$$\frac{k_B T}{m}\frac{d\rho}{dr} = -\frac{GM(r)}{r^2}\rho \tag{1.30}$$

or, rearranging:

$$\frac{r^2}{\rho}\frac{d\rho}{dr} = -\frac{Gm}{k_B T}M(r). \tag{1.31}$$

The $M(r)$ used here is the mass interior to the radius r. Using conservation of matter, we see that $M(r)$ obeys the relationship

$$\frac{dM(r)}{dr} = 4\pi r^2 \rho(r). \tag{1.32}$$

Combining this equation and eq.(1.31), we have

$$\frac{d}{dr}\left(\frac{r^2}{\rho}\frac{d\rho}{dr}\right) = -\frac{Gm}{k_B T}\left(4\pi r^2 \rho\left(r\right)\right) \tag{1.33}$$

or

$$\frac{d}{dr}\left(r^2 \frac{d\ln\rho}{dr}\right) = -\frac{Gm}{k_B T}\left(4\pi r^2 \rho\left(r\right)\right). \tag{1.34}$$

As was done in the previous section, we will now present a distribution function, and show that it leads to an equation similar to eq.(1.34), thus showing that it describes an isothermal model.

We take a distribution function of the form

$$f(\varepsilon) = \frac{\rho_1}{(2\pi\sigma^2)^{\frac{3}{2}}} e^{\varepsilon/\sigma^2}. \tag{1.35}$$

Integrating this function over all velocities, we obtain

$$\rho = \rho_1 e^{\frac{\Psi}{\sigma^2}} \tag{1.36}$$

which we can express in terms of Ψ:

$$\begin{aligned} \ln\rho &= \ln\rho_1 + \frac{\Psi}{\sigma^2} \\ &= \frac{\Psi}{\sigma^2} \end{aligned} \tag{1.37}$$

where $\ln\rho_1 << \frac{\Psi}{\sigma^2}$. Rearranging we have

$$\Psi = \sigma^2 \ln\rho \tag{1.38}$$

Now using Poisson's equation expressed in terms of the relative potential Ψ we have

$$\frac{1}{r^2}\frac{d}{dr}\left(r^2\frac{d\Psi}{dr}\right) = -4\pi G\rho. \tag{1.39}$$

We combine these two equations and obtain

$$\frac{1}{r^2}\frac{d}{dr}\left(r^2\frac{d}{dr}\left(\sigma^2\ln\rho\right)\right) = -4\pi G\rho \tag{1.40}$$

and rearranging we arrive at

$$\frac{d}{dr}\left(r^2\frac{d\ln\rho}{dr}\right) = -\frac{4\pi G}{\sigma^2}r^2\rho. \tag{1.41}$$

We note that eq.(1.41) is of the same form as eq.(1.34), with $\sigma^2 = \frac{k_B T}{m}$, and thus we have shown that the distribution function eq.(1.35) describes an isothermal model. We will now move on to lowered isothermal models, where we will use a variation of this distribution function.

Lowered Isothermal Models

An isothermal model describes a system in equilibrium. For this reason, it would seem to make intuitive sense that an isothermal model would adequately describe a globular cluster. Unlike the case of a single star in which power is constantly being produced, globular clusters do not have any (obvious) power production mechanisms, and one would think that the system should reach equilibrium quite rapidly. If this were the case, we would not venture to find more suitable models. However, globular clusters never reach equilibrium, and there are three evident mechanisms responsible for this behavior [1]:

1. The outer layer of a globular cluster is constantly being disrupted by tidal forces from its host galaxy, with stars being added to and lost from the cluster (mainly lost). This is accentuated by close encounters of stars giving some stars a very high kinetic energy.

2. The mass distribution changes constantly due to the fact that more massive stars are pulled closer to the core of the cluster. This is called the mass stratification instability.

3. There is indeed a power production mechanism: gravitational contraction. This leads to an increase in the magnitude of the potential, providing power, but is incompatible with equilibrium or a steady state.

The solution to the problems caused by these factors is to create modified isothermal models. These models resemble the isothermal sphere at small radii, but have a reduced density at large radii. This reduced density is imposed in an attempt to account for the first of these three mechanisms: it models the absence of certain stars with high kinetic energies which will have been stripped from the cluster by tidal forces. These models are the so-called lowered isothermal models.

One such modification of eq.(1.35) was introduced by I. King and is known therefore as the King model [1]. This lowered isothermal model was used in much of the work described in this thesis. The King model has the form

$$f(\varepsilon) = \begin{matrix} \frac{\rho_1}{(2\pi\sigma^2)^{\frac{3}{2}}}\left(e^{\frac{\varepsilon}{\sigma^2}} - 1\right) & \varepsilon > 0 \\ 0 & \varepsilon \leq 0\,. \end{matrix} \tag{1.42}$$

We can derive an equation for the density distribution of this model by integrating this distribution function over all velocities, and we obtain

$$\begin{aligned} \rho &= \frac{4\pi\rho_1}{(2\pi\sigma^2)^{\frac{3}{2}}}\int_0^{\sqrt{2\Psi}}\left[\exp\left(\frac{\Psi - \frac{1}{2}v^2}{\sigma^2}\right) - 1\right]v^2 dv \\ &= \rho_1\left[e^{\frac{\Psi}{\sigma^2}}\,\mathrm{Erf}\left(\frac{\sqrt{\Psi}}{\sigma}\right) - \sqrt{\frac{4\Psi}{\pi\sigma^2}}\left(1 + \frac{2\Psi}{3\sigma^2}\right)\right] \end{aligned} \tag{1.43}$$

where Erf(x) is the error function given by [17]

$$\mathrm{Erf}(x) \equiv \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}dt = \frac{2}{\sqrt{\pi}}\sum_{n=0}^{\infty}\frac{(-1)^n z^{2n+1}}{n!(2n+1)}. \tag{1.44}$$

Using eq.(1.43) we can express Poisson's equation for this system as

$$\frac{d}{dr}\left(r^2\frac{d\Psi}{dr}\right) = -4\pi G\rho_1 r^2 \left[e^{\frac{\Psi}{\sigma^2}}\,\mathrm{Erf}\left(\frac{\sqrt{\Psi}}{\sigma}\right) - \sqrt{\frac{4\Psi}{\pi\sigma^2}}\left(1+\frac{2\Psi}{3\sigma^2}\right)\right]. \tag{1.45}$$

To solve this equation, we choose our boundary conditions and integrate numerically.

The density of the King model decreases as r increases, until eventually the density becomes zero. This happens because at any given radius, the speed of stars must lie in the range $[0,\ \sqrt{2\Psi}]$, and thus there can be no stars at a radius at which $\Psi(r) = 0$. If we integrate eq.(1.45) outward, the initial conditions are $\frac{d\Psi}{dr} = 0$ and $\frac{d^2\Psi}{dr^2} < 0$. This means that at some value of r, $\Psi(r) = 0$ and therefore the density at this point is zero. The radius at which this occurs is known as the tidal radius.

It is important to note that the value of $\Psi(r)$ at $r = 0$ is not set for the King model. Using different values of $\Psi(r = 0)$ (which we will write as Ψ_0 henceforth) leads to different models. For the work described in this thesis the value $\Psi_0 = 5$ was used.

1.3 Computer simulations of globular clusters

There are many different algorithms that are used for numerical simulations of star clusters. Each of these algorithms implements what's known as an $N-$body simulation. In these simulations, every star in the cluster is treated as a point particle, and it is assumed that the only important force worth considering is the gravitational force.

1.3.1 $N-$body algorithms

Amoung the various different algorithms used to implement $N-$body simulations of globular clusters are the particle-particle method, the particle-mesh method, the treecode method, the fast multipole method and the symplectic method, just to name a few. The particle-particle method is the most basic of these, and it has a time complexity of $O(N^2)$. The faster methods, such as the treecode method, have a time complexity of $O(N \log N)$. The particle-mesh method is useful in special situations when softening of collisions between stars is desirable, and it has a time complexity of $O(N + N_g \log N_g)$, where N_g is the number of grid points used in the mesh. We note here that there is also a treecode method used in electrodynamics, however this is quite different from the $N-$body treecode method. From this point on when we discuss the treecode method, it is understood that we are are referring to the $N-$body algorithm.

In order for algorithms such as the treecode method to achieve a time complexity of less than $O(N^2)$ some approximations must be made. In the treecode method, this is done by clumping together groups of stars that are far from the particular star in question, and treating each of these groups as one big mass. This reduces the time it takes to calculate the force on each individual star at each step from $O(N)$ to $O(\log N)$, and thus the total time complexity of the treecode method becomes $O(N \log N)$.

Perhaps the best known treecode algorithm is the Barnes-Hut algorithm, and this is the algorithm which is implemented in the $N-$body code used in our research. The code we used is was written by J. Barnes, and the documentation for the code, along with the code itself, is available from [2]. Next we will give a description of how the Barnes-Hut algorithm works.

1.3.2 The Barnes-Hut algorithm

There are two basic steps that the Barnes-Hut algorithm uses to calculate the force on a particular star. The first step is the creation of a tree structure, which contains information about the positional layout of the stars, and the second step involves traversing this tree. First we will describe the process of creating the tree.

We start by calculating the greatest distance between any of the stars. We use this as the side length of a cube, which is centered at the center of the cluster, and thus contains all of the stars in the cluster. We then divide up this cube into octants, and check to see how many stars are in each octant. If there is more than one star in any of the octants, we recursively divide those octants up into eight more octants, repeating this procedure until every octant, known as "cells" [39], have either one or no stars in them. At each level of recursion, information about each cell is stored in a tree structure. Typically this information would be the center of mass and total mass of all the stars contained within the cell.

Once the tree is created, in order to calculate the force on a particular star we must "walk the tree". If we define D as the spacial extent of the cell and L as the distance from the center of the cell to the star in question, then we traverse the tree from the top node down, and at each node we do the comparison [39]

$$\frac{D}{L} < \theta \,, \tag{1.46}$$

where θ is a predefined parameter.

If $\frac{D}{L}$ is indeed less than θ then the current cell will be used to calculate the force on the star in question due to all stars contained within the cell. Otherwise, we continue down the tree until the comparison (1.46) becomes true. Once an acceptable node is found, we travel back up the tree to the last node which has a daughter node which wasn't used in a force calculation yet, and repeat the process [39].

This algorithm is used to calculate the force on a particular star due to every other star. This is done for every star at each timestep. In between these force calculations the implementation of this algorithm that we used in this thesis moves the stars using the leap-frog form of Verlet's method as the ordinary differential equation solver [2].

1.3.3 Important parameters

There are two parameters that affect the speed and accuracy of the Barnes-Hut treecode algorithm. The standard names for these parameters are *usequad* and *theta* [2].

The *usequad* parameter determines whether or not quad moments are used in the force calculation of the treecode. Turning it off (by setting its value as $false$) speeds up the calculation, but at the expense of some accuracy. For the most effective results, this parameter should be set to *true*, as the time saved is not worth the accuracy lost [13]. This parameter was set to *true* for all of the simulations used in this thesis.

The *theta* parameter determines the value of θ to be used in (1.46). Aarseth [13] suggests a value of 0.5 or 0.6 for the most effective accuracy/speed trade-off. A value of 0.5 was used in all of the simulations used in this thesis.

1.4 Motivation for research undertaken

1.4.1 Study of slowly-changing dynamical variables

When analyzing the behaviour of a physical system, we typically study variables such as position, velocity and acceleration. These variables describe the exact motion of particles in the system, from one moment to the next. In the case of periodic systems, however, it is sometimes of more interest to study variables which describe the changes in the periodic behaviour of particles over time. General properties of such a system are often easier to observe by analyzing these types of slowly-changing variables.

Such, indeed, is the programme followed in celestial mechanics, where it is usual, in calculating the gravitational interactions amoung planets, to formulate the calculations not in terms of relatively rapidly-changing variables such as orbital position and velocity, but in terms of orbital elements such as eccentricity which are more slow-varying. The best known of these are the Keplerian elements and the Delaunay variables [24].

The Delaunay variables are often used in studying astrophysical systems at the planetary scale. These are given by

$$l = n(t-\tau) \tag{1.47}$$

$$g = w \tag{1.48}$$

$$h = \Omega \tag{1.49}$$

and

$$L = \sqrt{\mu a} \tag{1.50}$$

$$G = \sqrt{\mu a(1-e^2)} \tag{1.51}$$

$$H = \cos i \sqrt{\mu a(1-e^2)} \tag{1.52}$$

where n is the mean motion, a is the semimajor axis, e is the eccentricity, i is the inclination, w is the argument of the perigee, Ω is the longitude of the ascending node and τ is the time when the satellite passes through the perigee [24]. The Delaunay variables make up a set of action-angle variables, where l g and h are the action variables, and L G and H are the angle variables. We note that L is related to the orbital energy of the two-body problem, while G is the magnitude of the orbital angular momentum and H is the Z component of the orbital angular momentum [3].

Though the Delaunay variables have been used with great success in the study of planetary systems, no such set of variables has been developed for the study of stellar systems. There is no obvious reason why study of stellar systems through slowly-changing variables similar to the Delaunay variables would not be of interest. For this reason, in chapters two and three of this thesis we will perform an ARMA time series analysis of two such variables.

The first of the variables that we will study is the theoretical period of the orbits of stars. We define the period of a star's orbit as the time it takes the particle to travel from $r_{\min}$ to $r_{\max}$ and back to $r_{\min}$, where $r_{\min}$ is the periapsis of the star's orbit, and $r_{\max}$ is the apoapsis. The period can be calculated using [32]

$$P = \sqrt{2m} \int_{r_{\min}}^{r_{\max}} \frac{dr}{\sqrt{E - U\left(r\right) - \frac{l^2}{2mr^2}}}. \tag{1.53}$$

The second slowly-changing variable that we will study is the angle between $r_{\min}$ and $r_{\max}$. The equation for this variable is [32]

$$\theta = \frac{l}{\sqrt{2m}} \int_{r_{\min}}^{r_{\max}} \frac{dr}{r^2 \sqrt{E - U\left(r\right) - \frac{l^2}{2mr^2}}}. \tag{1.54}$$

We will henceforth refer to this variable simply as the angle of a star.

In [4] we described the implementation and testing of a C program that takes the output of an $N-$body simulation and calculates the values of P and θ for each star at each timestep. This program was used to calculate the values of these variables for all of the work presented in this thesis.

1.4.2 Study of the force distribution of globular clusters

The force distribution of stars in a globular cluster is of great interest to astrophysicists. It is not a very active field of research, however, because it has been and still is generally believed that an appropriate such distribution has already been successfully derived. Professor Holtsmark went about this task in 1917 [34], and his result has been accepted without much question ever since then. As we will see in chapter 4 we discovered, quite by accident, that Professor Holtsmark's distribution is not equivalent to the real force distribution of a globular cluster. Obviously this prompted us to go about determining why this is the case, and to examine what is involved in deriving the correct force distribution.

1.5 Work done

In chapter 2 we perform an ARMA analysis of the period of several stars in an $N-$body simulation of a globular cluster. We describe in detail the procedure used, and then we present the results of our analysis. These results lead us to ask several questions, which we determine would be best answered by performing an automated ARMA analysis on *all* of the stars in a cluster.

In chapter 3 we outline how we designed our automated ARMA analysis. We then perform this analysis on both the period and the angle of every star in a number of different $N-$body simulations. We use our results, along with the results of some special test cases, to determine whether or not ARMA models can be successfully fit to these variables, and also to determine if an automated approach to ARMA analysis, which has not often been done before, has any merit.

We begin chapter 4 by presenting an expanded version of Chandrasekhar's famous re-derivation of the Holtsmark distribution [23]. We then describe how we went about testing the distribution, and we show that it does not correctly predict the distribution of forces in a globular cluster. We go on to show the problem with one of the assumptions made in the derivation, and then we demonstrate how the

Holtsmark distribution accurately predicts the force distribution only for stars at the center of a globular cluster. We conclude the chapter by presenting the beginnings of the derivation of a more generalized force distribution, which does not make the incorrect assumption made in the derivation of the Holtsmark distribution.

Chapter 2

ARMA analysis

2.1 Introduction

2.1.1 What is a time series?

A time series can be defined as any set of measurements made over a stretch of time or volume of space. Time series can be treated as continuous, but they are usually studied at discrete, evenly spaced intervals. Time series analysis of stochastic processes typically has two main goals. These are, first, to understand or model the stochastic mechanism that generates the series, and second, to predict future values of the series based on its past [28]. For the work done in this thesis we are only concerned with the former. It is worth noting that though most laws of physics are, in their common form, deterministic, in practice almost all processes are best modeled as stochastic (random) processes.

All of the work done in this thesis is on discrete time series, and it is common practice to use the variable N to refer to the number of discretization points [1]. We

[1] In Chapters 2 and 3, N will be the number of discretization points and N_* will be the number of stars in a cluster. In Chapter 1 above, however, N denoted the number of stars in a cluster, and

will define $X(t)$ as the value of the time series at time t, where t is an integer in the range 1 to N.

2.1.2 ARMA analysis

The AutoRegressive Moving Average (ARMA) models are a collection of very flexible time series models [31]. The process of determining what ARMA models, if any, fit a set of data is known as ARMA analysis. This process has four basic steps. First the data must be rendered stationary. Next a group of models that can be fit to this stationary data set is determined. The appropriate parameters for these models are then calculated. Finally the models, with the fitted parameters, undergo diagnostics where they are compared with the data to determine whether or not they are suitable.

2.1.3 Motivation

ARMA analysis of time series has been very successful in many different fields. It is of great interest in economics, with the emphasis being understandably given to its forecasting possibilities. It has found use in a variety of scientific fields, including hydrology, earth sciences, oceanography, marine biology, and even linguistics, where it has been used in modeling pronunciation networks [41]. This success makes any ARMA analysis that has not yet been attempted a very tempting research project. ARMA analysis of period and angle data from $N-$body simulations of globular clusters is one of these areas, and this is the analysis that will be presented in the next two chapters.

A successful ARMA analysis of this type would be the first step towards the derivation of a new set of kinetic equations describing globular cluster dynamics. The mo-

it will have the same usage in Chapter 4 tivation for such a set of equations is great — all one would need is the appropriate parameters for a particular cluster, and many important properties could be determined. The derivation of these equations, while it serves as motivation for this work, is itself beyond the scope of this thesis.

2.1.4 Chapter outline

We begin this chapter with an overview of time series analysis, with an emphasis on ARMA time series. We then discuss how to prepare the data for modeling, how to identify ARMA models that may potentially fit the data, and how to go about testing whether or not the predicted models are a good fit to the data. We go on to describe the procedure we used in doing an ARMA analysis of period data from a 1000 star simulation, and then we present the results of our analysis. We finish the chapter with a discussion of these results.

2.1.5 Software used

All of the time series analysis done for this thesis used the "Time Series Applications Package" for Mathematica®. Matlab® was used for some visualization work.

2.2 Stationarity requirement

In order to carry out an ARMA analysis, we require that the time series in question is stationary. What this means is that the probability laws governing the process do not change with time [28]. If, for example, the series shows a distinct trend, then it is the fluctuations about this trend that we wish to analyze. The full description of the data set can then be given by the transformations used on the raw data, and the model(s) that can be fit to the transformed data.

If we treat each value of a time series as a separate random variable, then each $X(i)$ has a probability density function $p(x_i)$. A set of $X(i)$ for i from m to $m+n$ has a joint distribution function $p(x_m, x_{m+1}, ..., x_{m+n})$. Formally, a set of data is deemed to be strictly stationary if this joint distribution function is independent of m for any positive integer n [14]. If this is the case the distribution function of the variable $X(t)$ is the same at each point, and the joint distribution function only depends on the spacing of the data points, not their values [31].

Second-order or weak stationarity of a set of data is achieved if the following two conditions are satisfied

$$1.\ \mu(t) = \mu \text{ and } \sigma^2(t) = \sigma^2 \text{ for all t} \tag{2.1}$$

$$2.\ \gamma(s, r) \text{ is a function of } (s - r) \text{ only,} \tag{2.2}$$

where $\gamma(s, r)$ is the covariance of $X(s)$ and $X(r)$. Usually only weak stationarity is required for the analysis of time series, and it is common practice to refer to weak "stationarity" simply as stationarity, as we will from now on.

In the simulations analyzed for this thesis there are no external forces acting upon the clusters. In this scenario, the period and angle of stellar orbits should, in theory, be stationary variables. We note at this point that any finite length time series may exhibit non-stationary behaviour due to short term effects, the most obvious example of which being the collision of two or more stars, which causes rapid deviations in the period and angle of those stars. Thus in order to analyze these variables over finite timescales we need to render their time series stationary.

2.3 Important models

2.3.1 White noise

The time series models that we will use require the use of normally distributed random variables. We define

$$a(t) = \sigma_a N(0,1)$$

where $N(0,1)$ is a normal random variable of mean 0 and variance 1, and $a(t)$ is an independent random variable associated with timestep t.

We can now construct a very simple type of time series model known as a normal white noise model. It is defined by

$$X(t) = a(t) \tag{2.3}$$

where t is an integer that ranges from 1 to N. This model is of theoretical usefulness, as we will see later. Note for a general white noise model $a(t)$ is not required to be normally distributed, though normally distributed white noise is by far the most commonly studied. We will refer to white noise models later on in describing residual testing, and in that case we will not make any assumptions as to the underlying distribution function of these models.

2.3.2 General linear models

The AR, MA and ARMA models that we will introduce later in the chapter can all be described by a general linear model, under the right conditions. Following ref. [28] we define a general linear model as a linear combination of an infinite number of past terms of a white noise model plus the present term

$$X(t) = a(t) + \sum_{i=1}^{\infty} \psi_i a(t-i). \tag{2.4}$$

2.3.3 Moving Average (MA) models

If the right hand side of eq.(2.4) is a finite series, i.e. if $\psi_i \equiv 0$ for all $i > j$ for some fixed j, then the model becomes a moving average model. Changing the notation from ψ to $-\theta$ we have

$$X(t) = a(t) - \sum_{i=1}^{q} \theta_i a(t-i). \tag{2.5}$$

Such a moving average model is referred to as an MA(q) model. The name of these models is derived from the fact that each successive $X(t)$ is generated by taking the set of weights 1, $-\theta_1$, $-\theta_2$, ... and moving them forward one unit of time [28]. A moving average model is equivalent to a general linear model with ψ_1 to ψ_q replaced by $-\theta_1$ to $-\theta_q$, and all ψ_i values with i greater than q equal to 0.

We define the moving average polynomial as

$$\Theta(x) = 1 + \sum_{i=1}^{q} \theta_i x^i \tag{2.6}$$

and the backspace operator B^i as

$$B^i X(t) = X(t-i). \tag{2.7}$$

We can now express eq.(2.5) in a more convenient, compact form:

$$X(t) = \Theta(B)a(t). \tag{2.8}$$

This is the standard form of MA(q) models found in most of the literature.

2.3.4 Autoregressive (AR) models

Autoregressive models are represented by

$$X(t) = a(t) + \phi_1 X(t-1) + \phi_2 X(t-2) + ... + \phi_p X(t-p). \tag{2.9}$$

Like moving average models, autoregressive models can be expressed in a compact form. We define the autoregressive polynomial as

$$\Phi(x) = 1 - \sum_{i=1}^{p} \phi_i x^i. \tag{2.10}$$

Using this, eq.(2.9) becomes

$$\Phi(B)X(t) = a(t). \tag{2.11}$$

An autoregressive model with p summation terms, such as eq.(2.10), is called an AR(p) model. Each term $X(t)$ is obtained from a weighted sum of past terms of $X(t)$ plus a normal random variable $a(t)$. Thus each term in the series is obtained by regressing through past terms — hence the name "autoregressive" model. The autoregressive model can be expressed in the form of a general linear model eq.(2.4), but only if it is stationary, the requirements for which will be discussed later.

2.3.5 Autoregressive Moving Average (ARMA) models

If we combine equations eq.(2.8) and eq.(2.11), we obtain an autoregressive moving average model of the form

$$\Phi(B)X(t) = \Theta(B)a(t). \tag{2.12}$$

These models are known as ARMA(p,q) models. Clearly, since AR and MA models can be expressed as general linear models, ARMA models can as well. An ARMA($p,0$) model is equivalent to an AR(p) model, and an ARMA($0,q$) model is equivalent to an MA(q) model. Thus we see that both AR and MA models can be considered as special cases of ARMA models.

2.3.6 Stationarity and invertibility conditions

AR models are not, by default, stationary. If certain stationarity requirements are not met by the ϕ values, then an AR model will quickly explode. To get an idea of why this happens, we'll take the example of AR(1):

$$X(t) = a(t) + \phi X(t-1). \tag{2.13}$$

If we assume that this process is stationary, then $\sigma^2_{X(t)} = \sigma^2_{X(t-1)} \equiv \sigma^2_X$. Calculating the variance we obtain:

$$\sigma^2_X = \phi^2 \sigma^2_X + 1. \tag{2.14}$$

Solving for σ^2_X we get

$$\sigma^2_X = 1/(1-\phi^2). \tag{2.15}$$

Clearly for the variance to be finite and non-negative we must have $\phi^2 < 1$. This is the stationarity condition for an AR(1) model. For the general case of an AR(p) model to achieve stationarity it is required that the zeros of the AR polynomial eq.(2.10) lie outside of the unit circle in the complex plane [5]. For the proof of this see ref. [14]. As was mentioned earlier, if an AR model is stationary, it can be expressed as a general linear model, as shown in [14] within the proof of the stationarity condition.

While all MA models are stationary, there is a set of conditions analogous to the stationary conditions for AR models that we require the θ values of an MA model to obey. The problem is that multiple MA models with different θ values can have the same auto-correlation function. The solution is to require that MA models are invertible.

An MA model is invertible if the zeros of the MA polynomial eq.(2.6) lie outside of the unit circle in the complex plane [5]. There is only one invertible MA model with a given auto-correlation function [28], and thus if we restrict ourselves to invertible MA models we solve this uniqueness problem. Also, an invertible MA model can be expressed as an AR(∞) model, just as the stationarity condition guarantees that an AR model can be expressed as an MA(∞) model (which is the same as a general linear model). This useful property of invertible MA models will be exploited later during model identification and fitting.

2.3.7 Autoregressive integrated moving average (ARIMA) models

If an ARMA model is non-stationary, then it will have one or more zeros of the autoregressive polynomial eq.(2.10) on or outside of the unit circle. However, if none of these zeros lie outside of the unit circle, but one or more of them lie *on* the unit circle, then we can transform the non-stationary series $X(t)$ into a stationary one using an operation known as differencing. Differencing is accomplished by applying the operator $1 - B$ to the series $X(t)$, where B is the backspace operator introduced in section 2.3.3. If we apply this operator d times, we obtain a new series

$$Y(t) = (1 - B)^d X(t). \tag{2.16}$$

Usually [28] differencing the series $X(t)$ once or twice is enough to obtain a stationary series $Y(t)$. We can then form a stationary ARMA model using the new transformed series $Y(t)$:

$$\Phi(B)Y(t) = \Theta(B)a(t). \tag{2.17}$$

Substituting eq.(2.16) in to this equation we obtain

$$(1 - B)^d \Phi(B)X(t) = \Theta(B)a(t). \tag{2.18}$$

These types of models are known as ARIMA(p, d, q) models, or autoregressive integrated moving average models.

The p and q values of an ARIMA(p, d, q) model are those of the ARMA model eq.(2.17), and d refers to the number of times the series $X(t)$ was differenced to obtain $Y(t)$. The values of p' and q' and the θ and ϕ parameters of the original, non-stationary ARMA model that describes $X(t)$ can be determined from the p, d and q values of this ARIMA model. The determination of p' and q' is simple: $p' = p + d$, $q' = q$. Obtaining the θ and ϕ parameters is not overly complicated, as is shown in [28].

It was mentioned earlier that we are only interested in stationary models, and thus the class of ARIMA(p, d, q) models encapsulates two necessary pieces of information in a compact form: the original series must be differenced d times to become stationary, and the resulting set of data can be fit to an ARMA(p, q) model.

2.4 Useful functions

There are four types of autocorrelation functions that are of great interest in time series analysis. The four types are the autocorrelation function, the partial autocorrelation function, the sample autocorrelation function and the sample partial autocorrelation function.

2.4.1 Definitions

The following definitions require that the time series is stationary. The autocorrelation function is defined by [28]

$$\rho_k = \frac{\gamma_k}{\sigma^2}, \tag{2.19}$$

where γ_k is the covariance of $X(t)$ and $X(t-k)$ and σ^2 is the variance of the time series.

The partial autocorrelation function is defined by [28]

$$\phi_{\mathrm{KK}} = Corr\,(\,(X(t), X(t-k)\mid X(t-1), X(t-2), \ldots, X(t-k+1)\,)\,. \tag{2.20}$$

Thus it is the autocorrelation function of $X(t)$ and $X(t-k)$ with all of data points in between treated as constants. Levinson and Durbin showed how this can be calculated, an explanation of which can be found in ref. [28].

The sample autocorrelation function is simply an estimate of the autocorrelation function of the data, and it is given by [28]

$$r_k = \frac{\sum_{t=k+1}^{N}(X(t)-\mu_X)(X(t-k)-\mu_X)}{\sum_{t=1}^{N}(X(t)-\mu_X)^2} \text{ for k = 0, 1, 2 ...} \tag{2.21}$$

The sample partial autocorrelation function is an estimate of the partial autocorrelation function of the data. We can obtain it using Levinson and Durbin's approach to calculating the real partial auto-correlation function, the only difference being that while the former calculation makes use of the autocorrelation function, the later uses the sample autocorrelation function in its place. This approach to calculating the sample partial autocorrelation function is know as the Levinson-Durbin algorithm. For more information regarding this algorithm see [6].

2.4.2 Usage

The main usage of the autocorrelation function in ARMA analysis stems from the fact that the autocorrelation function ρ_k of an MA(q) model is zero for $k > q$ [5]. While the sample autocorrelation function, which is what we use in practice, will not be exactly equal to zero in these cases, it can be shown that if

$$|r_k| < \frac{2}{\sqrt{N}} \tag{2.22}$$

then 95% percent of the time r_k is not statistically different from zero [5]. Thus we can calculate the sample autocorrelation functions for a number of different lags k and check to see if each of these obeys eq.(2.22). If one doesn't, then there's a good chance that an MA model of that order will fit the data.

The partial autocorrelation function has an analogous relationship with AR models. We check if

$$|\phi_{\mathrm{KK}}| < \frac{2}{\sqrt{N}} \tag{2.23}$$

for a number of values of k, and if this is false for any of them then an AR model of that order may fit the data.

2.5 Preparation of the data for modeling

Earlier we examined the requirement that data must be stationary before we can fit it to any ARMA models. From the definition of stationarity, eq.(2.1), it is clear that we must render the mean and variance of the series constant. How this was accomplished for the work done in this thesis is shown below.

2.5.1 Non-constant mean

Removing a non-constant mean (usually referred to as a trend) from a set of data is known as detrending the data, and this can be done in various ways. If we use a polynomial of degree d to describe the trend, then we can fit the parameters of the polynomial to the data, and then subtract the polynomial from the data to obtain the detrended series. While we wish to use this idea, differencing the data d times is equivalent to this method, and not only is it faster, it is also more convenient. If the transformed data can be fit to an ARMA(p, q) model, then the untransformed data will fit an ARIMA(p, d, q) model. As shown in section 2.3.7, ARIMA(p, d, q) models are equivalent to ARMA$(p+d, q)$ models, so using this method of detrending makes it easy to obtain an ARMA(p', q') model for the original data from the calculated ARMA(p, q) model obtained for the detrended data. This is the method used in this thesis to detrend data.

An example of the removal of a non-constant mean is shown in figures 2.1 and 2.2.

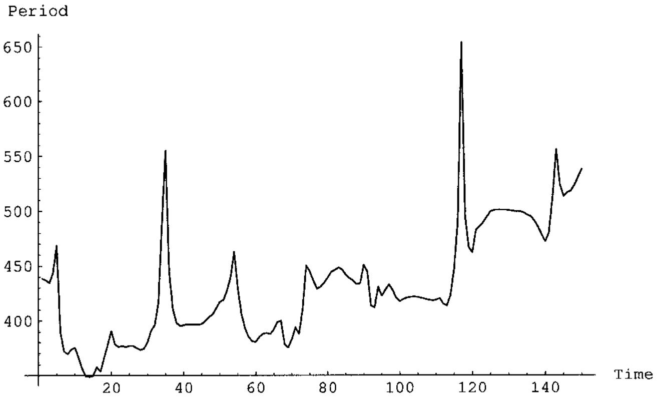


Figure 2.1: Plot of the period of a star, untransformed. We can see that it has an upward trend and a non-constant variance.

2.5.2 Non-constant variance

A non-constant variance is usually dealt with by using a non-linear transformation, such as a square root or a logarithm. For all the work done in this thesis a logarithm was used to remove a non-constant variance. An example of using a logarithmic transform to deal with a non-constant variance is given in figure 2.3.

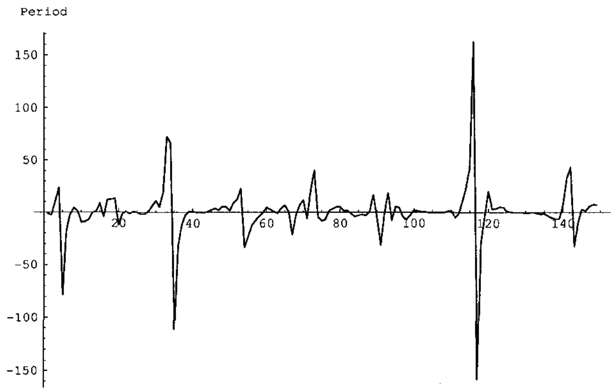


Figure 2.2: The period data from figure 2.1, differenced once. We can see that the upward trend is gone.

2.6 Model identification and fitting

2.6.1 Information criteria

Before we discuss the procedure used in this thesis for model identification and fitting, we will introduce two very useful information criteria. The first of these is known as Akaike's information criterion, or AIC, and it uses the statistic [5]

$$\ln(\hat{\sigma}^2) + \frac{2(p+q)}{n} \tag{2.24}$$

where $\hat{\sigma}^2$ is an estimate of the noise variance of a model. There are other definitions of this statistic, but we will use this one, following [5]. The second information criterion we will use is known as the Bayesian information criterion, or BIC, and it uses the statistic [5]

$$\ln(\hat{\sigma}^2) + \frac{\ln(n)(p+q)}{n}. \tag{2.25}$$

These two criteria are used to rank a number of different models which are all being fit to the same data. The smaller the statistic associated with the information criterion is, the better the model is considered to be.

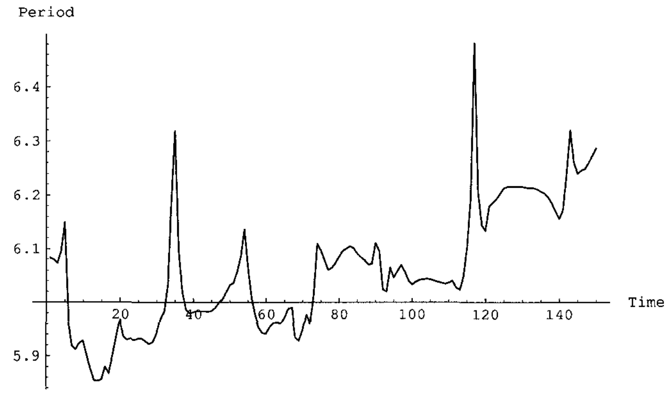


Figure 2.3: The period data from figure 2.1, with a log transform applied. There is little visible difference in the graphs, but we can see that the range is much smaller.

The first term in both of these functions can be thought of as a penalty for underfitting — fitting a model that does not have enough parameters. To see this, we note that if a model uses less than the necessary number of parameters, the estimates of the noise variance $\hat{\sigma}^2$ will be large, and thus $\ln(\hat{\sigma}^2)$ will be large. The second term in these functions can be thought of as the opposite of the first - a penalty for overfitting. The principle of parsimony, an important axiom of time series analysis, dictates that if two different models fit the data, but one has less parameters, then that is the best one to use. Simply put, the more terms that are used, the greater $(p+q)$ will be, and thus the larger the second term of eq.(2.24) and eq.(2.25) will get.

These two functions have been carefully designed to try and balance the need for enough parameters to model the data properly with the principle of parsimony.

2.6.2 The Hannan-Rissanen algorithm

There are many methods different algorithms that have been developed to identify suitable models for a set of data and estimate the parameters of these models. For the work done in this thesis we used the Hannan-Rissanen algorithm. This algorithm has the following steps [5]:

1. Use the Levinson-Durbin algorithm to fit the first $k_{\max}$ AR models to the data (see [5] for a full description of the Levinson-Durbin algorithm).

2. Of these $k_{\max}$ AR models, choose the one with the smallest AIC statistic.

3. Calculate the residuals obtained from fitting this AR model to the data and use these as estimates of the noise terms $\hat{a}(t)$ (for a definition of the residuals see section 2.7.1).

4. Use the least squares approach to estimate the parameters of every ARMA model with $p < \min(p_{max}, k)$ and $q < q_{max}$, where $p_{\max}$ and $q_{\max}$ are specified by the user.

5. Calculate the BIC statistics for all of these ARMA models and use this to rank them in order of preference.

An estimate of the variance of the noise for these models is given by [5]

$$\sum_{t=t'+1}^{N} \frac{\hat{a}(t)^2}{N - t'} \tag{2.26}$$

where $t' = Max(q + k, p)$. The fact that we only have $N - k$ noise estimates is the reason this is not summed over all N points. The restriction $p < Min(p_{\max}, k)$ in step four is in place for the same reason.

The implementation of the Hannan-Rissanen algorithm that is included with the time series add-on package for Mathematica® takes four parameters as input: $p_{\max}$, $q_{\max}$, $k_{\max}$ and h, where h is the number of models to be returned. Each time we used the Hannan-Rissanen algorithm we set $k_{\max} = 10$, which is more than enough to assure an appropriate AR model is chosen in step one. We chose appropriate values of the rest of the parameters depending on the usage.

2.6.3 Conditional maximum likelihood method

While the Hannan-Rissanen algorithm produces reasonable approximations of the parameters of a model, there are other algorithms designed to produce more reliable results when the values of p and q of the model are known. One of these is known as the maximum likelyhood method, and though it is quite effective, it is also extremely computationally time consuming [5]. An approximate version of this algorithm known as the conditional maximum likelyhood method also improves upon the parameter estimates given by the Hannan-Rissanen algorithm, and consumes much less time than the full maximum likelyhood method. We made use of the conditional maximum likelyhood method a number of times in our analysis. A description of the method is somewhat complicated, and can be found in refs. [33] or [44].

2.7 Model Diagnostics

There are various ways of testing the hypothesized models against the original data, the most common of which is residual testing [5]. If we take the difference of the original data and the predicted model, we get a set of data called the residuals (it should be noted that there are other definitions of the residuals, but it is this definition, taken from [5], that we will use). If the model is a good fit to the data, the residuals should behave like a white noise model with zero mean and constant variance ([5], [28]). It is easier to compare the behaviour of the residuals to a white noise model than it is to compare the behaviour of the original set of data to an arbitrary ARMA(p, q) model, and this is why residual testing is used.

There are many tests that can be used to determine if the residuals act like a white noise model [5]. Three of these tests were used in the work done for this thesis. These are the Portmanteau test, the turning points test and the difference-sign test. It should be emphasized before continuing that the purpose of these tests is to try and *disprove* the idea that the residuals have the properties of a white noise model. There is no way to provide positive proof that a data set is equivalent to a particular distribution, and so these tests try instead to disprove this idea. The more tests that fail to disprove an assertion, the more likely that assertion is correct, and for this reason we perform all three tests on all sets of residuals.

2.7.1 Portmanteau test

The Portmanteau statistic, also known as the modified Box-Pierce statistic, is given by [28]

$$Q_h = n(n+2)\sum_{k=1}^{h}\frac{r_k^2}{n-k}, \tag{2.27}$$

where r_k is the kth value of the sample autocorrelation function of the residuals. This test is designed to determine if the first h sample autocorrelation function values deviate significantly from zero, which would suggest that the residuals are not behaving as white noise. If the ARMA model in question is valid, then Q_h should have a distribution very similar to a chi-squared distribution with $h-p-q$ degrees of freedom [28]. Thus we can calculate Q_h, subtract the mean of $\chi^2(h-p-q)$ and determine within how many standard deviations of $\chi^2(h-p-q)$ Q_h lies. A value of h is required to do this test, and the value $h = 35$, suggested by [5], is the one used in the work done for this thesis.

This test as described above was implemented in a Mathematica® function PortmanteauTest as follows:

```
PortmanteauTest[res_, p_, q_, h_]:=
Block[{test, average, std},
test = PortmanteauStatistic[res, h];
average = Mean[ChiSquareDistribution[h − p − q]];
std = StandardDeviation[ChiSquareDistribution[h − p − q]];
Abs[test − average]/std
];
```

2.7.2 Turning point test

The turning point test is performed by counting the number of turning points that the residuals have, and comparing this value to the expected number. A turning point can be defined [5] as a data point $X(t)$ which satisfies

$$X(t) > X(t-1) \text{ and } X(t) > X(t+1) \tag{2.28}$$

or

$$X(t) < X(t-1) \text{ and } X(t) < X(t+1). \tag{2.29}$$

In other words, this data point is a local extremum. The number of turning points T is a random variable, and for a white noise model this variable has a mean [5]

$$\mu_T = 2\frac{N-2}{3} \tag{2.30}$$

and a variance

$$\sigma_T^2 = \frac{16N-29}{90} \tag{2.31}$$

where N denotes the length of the residuals (since they are the same length as the original series). T has an asymptotic normal distribution for a white noise model [5]. We use the same approach as with the Portmanteau test, that is we calculate the number of turning points T for the residuals, subtract the mean μ_T, and determine within how many standard deviations σ_T the value lies.

The Mathematica® function we wrote to implement this is given below:

```
TurningPointsTest[res_]:=
Block[{n, signs, test, average, std},
n = Length[res];
signs = Sign[Drop[res, 1] - Drop[res, -1]];
test = Count[Drop[signs, 1] * Drop[signs, -1], -1];
average = 2(n - 2)/3;
std = Sqrt[(16n - 29)/90];
Abs[test - average]/std
];
```

2.7.3 Difference-sign test

To do the difference-sign test, we count the number of times

$$X(t) - X(t-1) > 0 \tag{2.32}$$

for t from 0 to N, and we call this D. D is a random variable whose mean is

$$\mu_D = \frac{N-1}{2} \tag{2.33}$$

for a white noise model, as there should be an equal number of positive and negative differences between data points. The variance of the difference-sign test is [7]

$$\sigma_D^2 = \frac{N+1}{12}. \tag{2.34}$$

It should be noted that this is reported incorrectly by [5] to be $(n+1)/2$. Like T, D has an asymptotic normal distribution for a white noise model [5]. Once again, we calculate D for the residuals, subtract the mean, and determine the number of standard deviations that D is within.

The Mathematica® function used to do this test is shown here:

```
DifferenceSignTest[res_]:=
Block[{n, signs, test, average, std},
n = Length[res];
signs = Sign[Drop[res, 1] - Drop[res, -1]];
test = Count[signs, 1];
average = (n - 1)/2;
std = N[Sqrt[(n + 1)/12]];
Abs[test - average]/std
];
```

2.7.4 Evaluation of the residual testing procedures

Once the three residual tests were implemented in Mathematica® it was necessary to verify that they produced the expected results. First the turning points and difference-sign tests were run on a set of 100000 normally distributed numbers. Since normally distributed numbers are a normal white noise model, and T and D are asymptotically normally distributed for a white noise model, we would expect approximately 68.3

percent of T and D to fall within one standard deviation, 95.4 percent to fall within two standard deviations and 99.7 percent to fall within three standard deviations [8]. The results of these tests performed on time series of lengths 150, 250 and 1000 are shown in tables 2.1, 2.2 and 2.3 respectively.

Test	$1\,\sigma$	$2\,\sigma$	$3\,\sigma$
Difference-sign	74.1	95.2	99.8
Turning points	66.9	94.8	99.7

Table 2.1: Percentage passes of Difference-sign and Turning points tests on white noise, $N = 150$

Test	$1\,\sigma$	$2\,\sigma$	$3\,\sigma$
Difference-sign	72.7	95.3	99.8
Turning points	67.5	94.9	99.7

Table 2.2: Percentage passes of Difference-sign and Turning points tests on white noise, $N = 250$

Test	$1\,\sigma$	$2\,\sigma$	$3\,\sigma$
Difference-sign	67.6	95.2	99.7
Turning points	67.0	95.2	99.7

Table 2.3: Percentage passes of Difference-sign and Turning points tests on white noise, $N = 1000$

As we can see, the results were very good, with the exceptions of the difference-sign test for $n = 150$ and $n = 250$, where the number of test values falling within one standard deviation was slightly higher than expected. This is due to the fact that D, the value calculated in this test, has an *asymptotic* normal distribution. The results for two and three standard deviations, however, were very close to the expected values, and since we did not reject a model unless one of the test values was greater than two at the very least, this slight difference for one standard deviation was of no consequence.

Evaluation of the Portmanteau test was more difficult. When this test is used on a set of residuals, it takes as input the p and q values of the model being fit to the data. If we generated a realization of a particular ARMA model, and then fitted this model to the generated data, the residuals resulting from this fit should pass the Portmanteau test; that is to say the distribution of Q_h should behave similar to $\chi^2(h - p - q)$. Since we used a value of 35 for h, the chi-squared distributions we dealt with behaved almost identically to the normal distribution. The expected percentage of instances of these chi-squared distributions falling within one, two and three standard deviations was within 0.5% of what we would expect for the normal distribution, and thus the results of this test were comparable to the results shown in tables 2.1, 2.2 and 2.3.

We also performed the turning points and difference-sign tests on these residuals, and we determined how often the residuals passed all three tests. Cut-off points of two and three standard deviations were used. The higher the percentage was in each of these cases, the fewer good model fits would be rejected incorrectly if this number of standard deviations was to be used as a cut-off point.

For this test we used seven different ARMA models. These models were: AR(0.2), AR$(0.34, 0.52)$, AR$(0.12, 0.1, 0.34)$, MA(0.55), MA$(0.48, 0.82, 0.12)$, ARMA$(\{0.3, 0.23\}, \{0.6\})$ and ARMA$(\{0.17, 0.22, 0.41\}, \{0.15, 0.23\})$. The results of all the tests were then averaged and they are shown in tables 2.4,2.5 and 2.6. These tables show the results when 500, 1500 and 5000 instances of each model were used. Each time, like above, time series lengths of 150, 250 and 1000 were tested.

N	**Portmanteau**	**Portmanteau**	**Portmanteau**	**All tests**	**All tests**
	1σ	2σ	3σ	2σ	3σ
150	62.6	89.0	95.8	81.7	95.4
250	62.1	89.8	96.7	81.4	96.5
1000	64.7	92	98.4	83.4	97.9

Table 2.4: Average percentage passes of Portmanteau test and all three tests on 500 instances of seven different ARMA models

N	**Portmanteau**	**Portmanteau**	**Portmanteau**	**All tests**	**All tests**
	1σ	2σ	3σ	2σ	3σ
150	61.7	89.0	96.5	81.7	96.3
250	63.8	91.0	97.3	82.9	96.5
1000	65.0	91.7	98.1	83.5	97.6

Table 2.5: Average percentage passes of Portmanteau test and all three tests on 1500 instances of seven different ARMA models

N	Portmanteau	Portmanteau	Portmanteau	All tests	All tests
	1σ	2σ	3σ	2σ	3σ
150	62.4	89.3	96.3	81.3	96.0
250	63.9	91.9	97.1	82.4	96.7
1000	65.3	93.0	98.5	84.0	97.7

Table 2.6: Average percentage passes of Portmanteau test and all three tests on 5000 instances of seven different ARMA models

We note that the Portmanteau test does not quite perform as well as hoped in all of these cases, this being a result of the fact that this statistic only *approximately* behaves as a chi-squared distribution. As a consequence of this, the pass rate would be $< 85\%$ if two standard deviations was used as a cut-off. In contrast, the pass rate would be $> 95\%$ if three standard deviations was used as a cut-off. Clearly this indicates that an estimated model should not be automatically rejected simply because one of the residual tests turned out greater than 2, but if any of the tests had results greater than three it would be reasonable to reject the model.

It is also noteworth that the results are very similar for each different number of instances used, indicating that 500 instances is a reasonable sample size when the model type is known. Also, the difference in the results for the different values of the time series length N is minimal, since $N = 150$ has only slightly worse overall pass rates than $N = 1000$.

2.8 Smoothing of the spectrum

In order to compare the calculated spectrum of a data set with the spectra of various models we need to do some type of smoothing. For the spectral analysis done in this work we used a simple Daniell window — a rectangular window with equal weights defined by [5]

$$W(k) = \begin{cases} \frac{1}{2M+1}, & \text{for } |k| <= M \\ 0, & \text{otherwise} \end{cases} \tag{2.35}$$

Chatfield [25] suggests using an M value of approximately $M = N/40$, experimenting with values slightly bigger and smaller than this, and this is what we did.

2.9 Description of procedure

The first results we obtained were for the periods of individual stars (analysis of angle data will be shown in the next chapter). These stars were sorted according to their initial radius, with star 1 being the closest to the center, and star N_* being the farthest from the center, where N_* is the number of stars in the simulation. For all of the individual analysis, we used a cluster of size $N_* = 1000$.

Each analysis followed the same basic procedure. First, we plotted the data, and visually determined if it was stationary. If it was not, we differenced the data until we obtained a time series with a constant mean, and we applied a log transformation to get rid of any non-constant variance. Next we inspected plots of the first thirty values of the sample autocorrelation function and sample partial autocorrelation function. If any points within the first ten or so were located outside of the bounds given by eq.(2.22) and eq.(2.23), this indicated that an AR model, in the case of the sample autocorrelation function, or an MA model for the sample partial autocorrelation function may have been an appropriate fit to the data. If points greater than ten were located outside of the bounds, this indicated that the data may have been in need of seasonal differencing.

Next we used the Hannan-Rissanen algorithm to obtain a list of six models that could possibly fit the data. We calculated the AIC and BIC statistics of these models, and we used the results of the sample autocorrelation function and sample partial autocorrelation function test described above, along with the AIC and BIC statistics of the six models provided by the Hannan-Rissanen algorithm, to pick three models that were likely to fit the data. Once we had picked these models, we used the conditional maximum likelyhood method to obtain better estimates of their parameters.

We then performed a residual analysis on each model individually. We started the residual analysis the same way we started the analysis of the original series — we visually inspected a plot of the residuals. If there was no obvious trend or non-constant variance present, then we calculated and plotted the autocorrelation function of the residuals, and compared this to the cutoff given by eq.(2.22). In 95% of cases the values of the sample autocorrelation function of a white noise process should be within this bound. Thus if one or more of the r_k values was outside the bound this was an indication that the residuals were not acting like a white noise process, and if this was the case then the model in question was not a good fit. However, a model usually can't be eliminated purely on this basis, and thus unless many r_k values were outside of the bound, we needed to do more tests.

The next step was to apply the residual tests discussed in section 2.7. If the results of all three tests were within two standard deviations, then the model passed the residual tests. Test results of between two and three standard deviations may have been acceptable, depending on how many of them were in this range and whether or not any values of r_k were located outside of the bounds. Whether or not a model should be rejected in this situation is a complicated matter, and depends greatly on the exact results. If any of the test results were greater than three standard deviations, the model was rejected.

If the residual tests passed, the final step was to calculate and smooth the spectrum of the data, and compare it to the spectrum of the model in question. A visual inspection of these graphs placed on the same plot was used to determine if the spectrum was a reasonable fit. We then combined all of the infomation gathered to determine whether to accept or reject the model.

2.10 Results of individual analysis

Nine stars were picked at random from a cluster of size $N_* = 1000$, to be subjected to individual analysis. The stars picked were numbers 7, 107, 273, 391, 395, 406, 883, 898 and 928, where the stars were sorted by their distance from the cluster center. The results of the various stages of analysis are shown in table 2.7 at the end of the chapter. A key for the column headings of table 2.7 is given in table 2.8, and the fitted parameters of each model tested are shown in table 2.9.

2.10.1 Analysis of results

When we look at the results of these individual analysis, we note a number of things. First of all, in each case that worked it was found that one log transform and one differencing of the data was all that was required to render the data sufficiently stationary. In cases that didn't work, two and three differencings were attempted, both with similar negative results. All of the results shown in table 2.7 were obtained by applying one log transform to the data, and then differencing the data once.

The second thing worth mentioning is that the models suggested by the points that lay outside of the correlation bounds were mostly quite different from the models that were predicted by using the Hannan-Rissanen algorithm combined with an evaluation of the AIC and BIC statistic of each model. This could be in part due to the varying degree to which points were located outside of the bounds, since the farther outside the bounds the outlying points lay, the more likely the model they predicted would work.

Next we consider the models that were chosen for each star. These were mostly quite low order, with AR(1) and MA(1) being the most common by far, AR(1) being chosen six times and MA(1) being chosen five times. Of course these lower order models are preferentially chosen by the information criterion, and thus it isn't clear from these results whether or not higher order models would work very often.

We can see that when a star can be fit, it can be fit to multiple different models. As we just mentioned, lower order models such as AR(1) and MA(1) are preferred by the information criterion, and since the AIC and BIC statistic are what we used to determine which models to try and fit, fitting of these models was attempted more often than other models. However it seems that they are not the only models that fit the data in these situations, indicating that some stars may simply be easy to fit to any model, while others are not. A clear question that arises from this is whether or not for these stars higher order models could also be successfully fit.

The predicted parameters of the models were distributed over a wide range, varying from less than 0.1 to over 0.5. Models that worked for multiple different stars such as AR(1) and MA(1) had widely varying parameters each different time they were successfully fit.

It is quite noticeable that as we look at the results of the residual tests and the spectral analysis, the spectra are mostly great or at least reasonable fits, which does not correspond with the results of the residual testing. A number of the stars that fail the residual testing do well in the spectral analysis, the most notable example of this being star 883, which despite having very nice spectral fits has very bad residual testing results. An example of the spectral fit for the ARMA$(1,2)$ model fit to star

883 is given is figure 2.4, and we can see that the spectrum of the model fits the data extremely well.

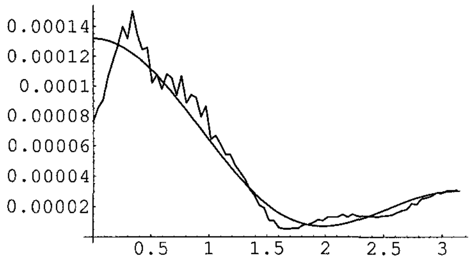


Figure 2.4: Smoothed spectral data and the spectrum of an ARMA($\{-0.00379617\}, \{0.533336, 0.499463\}$) model for star 883

A number of interesting questions arise from these results. Does transforming the data by applying a log transform and differencing it once work best for all of the stars? How many models can be fit to each data set that can be fit to any models? How often do various models fit the data sets? One way to find an answer to these questions was to do an analysis of data from a large group of stars. The easiest way to do this was to automate the ARMA analysis, and how this was done is the subject of our next chapter.

Star	OCB	OPCB	Best 3	Port	Turn	Diff	Spect	Fit?
7	MA(1)	None	AR(1)	1.40	0.78	1.13	Great	Accept
			ARMA(1,1)	1.41	0.39	1.41	OK	Accept
			MA(2)	0.87	0.39	1.13	Great	Accept
107	None	None	MA(1)	1.16	1.56	1.41	OK	Accept
			AR(1)	1.16	1.56	1.41	OK	Accept
			ARMA(1,1)	1.30	1.56	1.41	OK	Accept
273	MA(1)	None	AR(2)	1.06	0.20	0.57	OK	Accept
			AR(3)	0.43	0.20	1.13	Great	Accept
			MA(1)	2.04	1.37	0.28	OK	Accept
391	MA(1)	None	MA(3)	2.43	2.93	1.13	Great	Border
			MA(4)	2.46	2.93	0.0	Great	Accept
			ARMA(1,3)	2.04	2.54	1.70	Great	Accept
395	None	AR(4), AR(5)	AR(1)	1.06	2.54	2.26	OK	Accept
			AR(2)	0.31	2.54	3.39	OK	Reject
			MA(1)	1.22	2.54	2.26	OK	Accept
406	MA(1)	None	AR(1)	0.58	2.35	1.13	OK	Accept
			MA(1)	0.29	1.56	1.41	OK	Accept
			ARMA(1,1)	0.70	1.95	1.13	OK	Accept
883	MA(1)	AR(2)	MA(2)	2.96	9.19	3.96	Great	Reject
			ARMA(1,2)	2.87	9.19	3.68	Great	Reject
			AR(3)	2.62	10.36	2.83	Great	Reject
898	MA(6)	AR(5)	AR(1)	1.97	6.84	5.09	Bad	Reject
			AR(2)	1.04	6.06	3.39	OK	Reject
			AR(3)	0.99	6.45	3.68	Great	Reject

Star	OCB	OPCB	Best 3	Port	Diff	Turn	Spect	Fit?
928	MA(1)	None	AR(1)	1.73	0.98	0.28	Great	Accept
			MA(1)	1.45	0.59	0.28	Great	Accept
			ARMA(1, 1)	1.62	0.98	0.28	Great	Accept

Table 2.7: Results of ARMA analysis of nine different stars

Star = Star number
OCB = Suggested by point outside of autocorrelation function bounds
OPCB = Suggested by point outside of partial autocorrelation function bounds
Best 3 = Best three models
Port = Result of Portmanteau test
Diff = Result of Difference-sign test
Turn = Result of Turning points test
Spect = Quality of spectral fit
Fit? = Whether we decided to accept or reject the model

Table 2.8: Key for table 2.7

Star	Model	Fitted parameters
7	AR(1)	0.35
	ARMA(1, 1)	{0.57}, {-0.26}
	MA(2)	{0.31, 0.19}
107	MA(1)	1.8 E-2
	AR(1)	1.9 E-2
	ARMA(1, 1)	{3.1 E-2}, {-1.3 E-2}
273	AR(2)	{0.21, 0.10}
	AR(3)	{0.22, 0.13, -0.17}
	MA(1)	0.19
391	MA(3)	{0.48, 0.18, -4.0 E-2}
	MA(4)	{0.48, 0.17, -6.3 E-2, -4.8 E-2}
	ARMA(1, 3)	{-0.11}, {0.61, 0.17, 1.5 E-2}
395	AR(1)	0.15
	AR(2)	{0.13, 0.15}
	MA(1)	0.13
406	AR(1)	0.27
	MA(1)	0.26
	ARMA(1, 1)	{0.25}, {1.3 E-2}
883	MA(2)	{0.52, 0.50}
	ARMA(1, 2)	{-3.8 E-3}, {0.53, 0.50}
	AR(3)	{0.52, 0.20, -0.32}
898	AR(1)	-3.1 E-2
	AR(2)	{-2.5 E-2, 0.13}
	AR(3)	{-3.5 E-2, 0.13, 9.2 E-2}

Star	**Model**	**Fitted parameters**
928	AR(1)	0.31
	MA(1)	0.30
	ARMA(1, 1)	{0.29}, {1.8 E-2}

Table 2.9: Fitted parameters for the models from table 2.7

Chapter 3

Automation of ARMA Analysis

3.1 Introduction

3.1.1 Motivation

Many ARMA analysis can be done effectively by studying only 10 or 20 sets of data (for example [42] and [30]), and in this situation there is a clear advantage to examining the data by eye and doing a careful individual analysis. We did this in the last chapter, only to find that we could draw no clear conclusions. Many of the questions which were posed in the end of the last chapter would be very difficult to answer by performing more individual analysis, because a very large number of analysis would be required. An automated approach, such as the one we describe in this chapter, turns out to be an effective tool in answering these questions.

This thesis is not the first academic work to suggest automating an ARMA analysis. Professor Broersen first suggested the idea in ref. [18], and has since expanded on it in several papers (refs. [21], [19], [20], [22]). In ref. [20] he outlines some basic criteria involved in fitting time series models to data in an automated way, and then discusses an algorithm that he has developed to perform automated time series analysis. He says "As far as the author is aware, the algorithm of Section VI is the first successful attempt to automatically identify a time series model for measured observations without interaction from the user" [20], indicating that this is a very new idea. His five papers on the subject have been cited by other publications at least 71 times [9] to this date, indicating general interest in professor Broersen's method.

The algorithm that Professor Broersen uses, which he named "ARMAsel", picks out a single ARMA model that best fits the given data. We can clearly see from the results of the last chapter that there is no one particular model which will fit the period of all of the stars in a cluster. Thus our approach was designed to determine how many times each ARMA model (of the 35 types considered) could be successfully fit to our data, and thus try and conclude which model(s), if any, best describe the behaviour of the period and angle of all of our stars. We also used our automated script to compile statistics which could be used to answer many of the questions we have posed thus far.

Ultimately there were two fundamental questions we looked to answer by performing this automated ARMA analysis. The main question was, simply put, can ARMA models be used to effectively model period and angle data for stars in a globular cluster? A secondary question we considered addresses the usefulness of the approach itself — is an automated ARMA analysis of a large number of time series an effective means by which to determine if ARMA models can be fit to this data? We will address both of these queries again at the conclusion of the chapter.

3.1.2 Chapter outline

We begin with a description of how we classified the data. Next we discuss the the design of the automated procedure. We go on to explain how we tested our automated script, and then we provide the results of our analysis applied to period and angle data using time series of lengths $N = 150$, $N = 250$ and $N = 1000$. After showing that no trends were present in the data, we apply a number of test cases to our automated script, to determine its behaviour under controlled conditions. Using the outcome of these test cases we interpret our results. We conclude by discussing possibilities for further work.

3.2 Classifying the data

We took advantage of the benefits of an automated procedure to add a classification of the data. A plot of the period or angle of a typical star is composed of small perturbations interspersed with occasional large jumps due to close encounters with other stars. We decided that for the automated analysis we should divide the stars in each cluster into two categories: stars with large jumps and stars without large jumps.

In order to determine whether or not a particular time series contains large jumps we used the following method. First, we divided the time series up into ten segments. Next we calculated the standard deviation of each of these segments. We took the mean of these ten standard deviations, and then compared each standard deviation to this mean. If any of these local standard deviations was significantly greater than the mean, this meant that this portion of the time series changed more rapidly than the rest, indicating a high probability that this star has collided with another star during this period of time.

While performing the above procedure using $N - size_{window}$ points would provide more accurate results, this would take too long to do a large number of times. The above method cuts down on the number of standard deviation calculations that are required and provides reasonable results.

The brunt of our analysis was performed on stars which were accepted by this classification procedure, however we briefly examine the rejected stars in section 3.6.1.

3.3 Parts of the individual analysis that were removed

There were three elements of the individual analysis that were not incorporated in the automated approach. First of all, the analysis of the autocorrelation function and partial autocorrelation function was not very helpful in the individual analysis, and automating this process effectively would be quite challenging. Thus it was decided that this portion of the individual analysis would not be included in our automated script.

As we discussed in chapter 2, the results of the spectral analysis seemed to be much more accepting than the residual tests, and thus this analysis did not contribute a great deal to our conclusions. Spectral analysis is typically accomplished using the mark-one eyeball and a degree of human intuition, and thus, as with the analysis of the correlation functions, it is not an easy process to automate. For these reasons the spectral analysis was also cut from our automated script.

The last element we did not incorporate in the automated script was the analysis of the AIC statistic of models picked by the Hannan-Rissanen procedure. The reason this was not included is simple — we no longer chose from a number of models predicted by the Hannan-Rissanen algorithm, rather we decided to perform residual tests on all of the models that the Hannan-Rissanen algorithm chose! We set the total number of models picked by this algorithm to ten, which assured that all the models that were most likely to fit the data were considered.

3.4 Remainder of the procedure

In each of the individual star analysis performed in chapter 2, the raw data were non-stationary. Each time we were able to render the data stationary by differencing the data once and applying a logarithmic transformation. We assumed that this method would be acceptable for all of the stars, and thus we used it in the automated analysis. We re-examine this assumption in section 3.9, and again in section 3.12.2. After rendering the data stationary, we subtracted the mean of the data from the data.

As was done in the individual analysis, the Hannan-Rissanen algorithm was used to determine a suitable set of models along with their parameters. Earlier we posed the question "How many models can be fit to each data set?" In order to answer this question, we decided to increase the number of models that the Hannan-Rissanen algorithm returned from six to ten. The value of $k_{\max}$ was kept at ten, as mentioned above, and the values of $p_{\max}$ and $q_{\max}$ were set to five.

Next we performed the residual tests described in section 2.7 on each model predicted by the Hannan-Rissanen algorithm. If the results of all three tests each fell within three standard deviations, we accepted the model. Each time a model passed the residual tests we stored the model type and each of the fitted parameters.

Since $p_{\max}$ and $q_{\max}$ were set to five, as mentioned above, this meant that there were five AR models, five MA models and 25 ARMA models being tested. AR(p) models are equivalent to ARMA($p, 0$) models and MA(q) models are equivalent to ARMA($0, q$) models, and thus the easiest way to present the results was with a table consisting of six rows and six columns, with $p = 0$ through $p = 5$ as the rows and $q = 0$ and $q = 5$ as the columns. The table gave the percentage of times that each model was identified by the procedure and passed the residual tests. Thus the percentage of each AR model that was successfully fit was given in the first column, the MA results were placed in the first row, and the rest of the table showed the percentage of successful fits for the other ARMA models.

Along with this table eight statistics were calculated and displayed. We describe these statistics below, along with the abbreviations we will use to represent each one when we present the results.

- The number of stars that left the cluster during the simulation (starsLeft)
- The number of stars that were rejected due to large jumps (starsRejected)
- The number of stars that were analyzed (starsAnalyzed)
- The percentage of stars for which at least one model could be fitted (atLeastOne)
- The average number of models that were successfully fit, ignoring stars which could not be fit to any models (Average)
- The percentage of times the best ten models for a given star all failed the Portmanteau test (failedPort)
- The percentage of times the best ten models for a given star all failed the turning points test (failedTurn)

- The percentage of times the best ten models for a given star all failed the difference-sign test (failedDiff)

Stars that left the cluster could not be analyzed, due to the fact that they were no longer orbiting the center and thus did not have a meaningful period or angle. Clearly the number of stars analyzed was given by $N_* - starsLeft - starsRejected$.

3.5 Testing of the automated script

In order to assure that the automated script was functioning correctly, we tested it using the nine stars that were analyzed in the previous chapter. For each star we obtained the results of the residual tests applied to the seven models that we did not test in our individual analysis. We then ran each star through the script individually, and we compared the results of the model fitting to the results of our manual calculations. In each of the nine cases the results were exactly the same.

Next we manually compiled the total percentage of fits of each ARMA model for these nine stars. We then ran the nine stars through the script, and compared the results. Again, the results were exactly the same, proving that our script gives correct results.

3.6 Results for $N = 150$

The initial results were obtained using time series of length $N = 150$. Period and angle data for clusters of size $N_* = 1000$, $N_* = 3000$ and $N_* = 10000$ were analyzed. At least three simulations of each size were used. The only difference in the simulations was the initial conditions — for each simulation a different instance of the same $\Psi_0 = 5$ King model was used. All simulations of equal size produced roughly the same results, the results being more equivalent as the simulation size increased, for obvious reasons. This was true for all of the results we obtained, and for this reason we will present one set of results per cluster size.

We give the statistics calculated by the script for period data in table 3.1, and the statistics for angle data are presented in table 3.2.

N_*	starsLeft	starsRejected	starsAnalyzed	atLeastOne	Average	failedPort	failedTurn	failedDiff
1000	41	524	435	63.4%	5.6	0.0%	32.4%	11.5%
3000	77	1395	1528	75.7%	6.1	2.2%	12.0%	14.3%
10000	33	4802	5165	91.8%	7.4	0.8%	6.5%	1.7%

Table 3.1: Statistics for star periods, $N = 150$

N_*	starsLeft	starsRejected	starsAnalyzed	atLeastOne	Average	failedPort	failedTurn	failedDiff
1000	41	309	650	52.0%	6.0	1.4%	36.9%	10.3%
3000	77	758	2165	76.2%	6.9	0.4%	21.2%	5.6%
10000	33	2436	7531	84.1%	7.2	0.4%	14.0%	4.0%

Table 3.2: Statistics for star angles, $N = 150$

The first thing we notice when analyzing these results is that about 20% more of the angle data are accepted by our classification scheme as compared to the period data. It seems that the angle data are less affected by collisions. Other than this difference, the angle and period data seem to behave in a similar fashion.

A number of trends can be spotted in these results. The number of stars that can be fit to at least one model gets larger as the cluster size goes up. The same is true for the average number of models that can be fit to these particular stars. Clearly the bigger the cluster, the more successfull the ARMA analysis is. For all cluster sizes this average was greater than five. This result is significant, and we will refer to it below.

Next we present the results of the model fitting in tables 3.3 through 3.8.

	q=0	q=1	q=2	q=3	q=4	q=5
p=0		38.9	42.1	45.7	37.9	36.1
p=1	34.5	35.6	20.2	9.7	2.5	1.1
p=2	11.7	12.9	4.6	0.7	0.9	0.5
p=3	7.8	4.6	0.5	0	0.5	0.7
p=4	3.7	1.1	0.0	0.0	0.2	0.0
p=5	0.9	0.7	0.0	0.2	0.0	0.0

Table 3.3: Percentage of star periods that can be fit to ARMA(p,q) models, with $N_* = 1000$, $starsAnalyzed = 524$, $N = 150$

	q=0	q=1	q=2	q=3	q=4	q=5
p=0		43.2	40.9	41.7	37.4	36.6
p=1	36.0	39.4	23.8	13.5	4.9	0.8
p=2	11.2	10.5	4.5	0.2	0.9	0.6
p=3	4.8	4.0	0.0	0.2	0.3	0.3
p=4	3.1	0.6	0.2	0.0	0.2	0.2
p=5	0.9	0.2	0.2	0.0	0.0	0.0

Table 3.4: Percentage of star angles that can be fit to ARMA(p,q) models, with $N_* = 1000$, $starsAnalyzed = 650$, $N = 150$

	q=0	q=1	q=2	q=3	q=4	q=5
p=0		47.0	49.7	50.3	42.7	37.0
p=1	46.7	46.4	30.0	15.0	5.4	1.6
p=2	19.2	18.7	8.8	1.4	1.0	0.8
p=3	13.2	9.2	1.2	0.3	0.1	0.5
p=4	6.3	2.4	0.2	0.1	0.1	0.3
p=5	2.2	0.4	0.1	0.1	0.1	0.2

Table 3.5: Percentage of star periods that can be fit to ARMA(p, q) models, with $N_* = 3000$, $starsAnalyzed = 1528$, $N = 150$

	q=0	q=1	q=2	q=3	q=4	q=5
p=0		61.6	61.1	60.3	52.8	49.3
p=1	59.0	57.1	38.1	17.3	5.1	0.9
p=2	16.5	15.0	7.1	0.8	0.2	0.2
p=3	8.6	4.4	0.4	0.1	0.1	0.1
p=4	4.5	0.4	0.1	0.0	0.0	0.0
p=5	1.5	0.2	0.0	0.0	0.0	0.0

Table 3.6: Percentage of star angles that can be fit to ARMA(p, q) models, with $N_* = 3000$, $starsAnalyzed = 2165$, $N = 150$

	q=0	q=1	q=2	q=3	q=4	q=5
p=0		77.1	78.4	77.7	62.7	53.7
p=1	75.8	70.2	44.5	18.3	5.1	1.8
p=2	30.8	27.4	12.5	1.7	1.0	0.6
p=3	17.3	8.8	0.8	0.2	0.2	0.2
p=4	9.3	2.1	0.3	0.0	0.1	0.2
p=5	2.7	0.4	0.0	0.0	0.0	0.0

Table 3.7: Percentage of star periods that can be fit to ARMA(p, q) models, with $N_* = 10000$, $starsAnalyzed = 5165$, $N = 150$

	q=0	q=1	q=2	q=3	q=4	q=5
p=0		71.4	70.9	70.9	62.4	56.7
p=1	70.5	67.5	43.1	18.1	4.8	1.0
p=2	19.2	17.7	8.4	0.7	0.3	0.3
p=3	9.7	5.2	0.5	0.1	0.1	0.2
p=4	4.8	0.8	0.1	0.0	0.0	0.1
p=5	1.5	0.3	0.1	0.0	0.0	0.0

Table 3.8: Percentage of star angles that can be fit to ARMA(p, q) models, with $N_* = 10000$, $starsAnalyzed = 7531$, $N = 150$

We can see from observation of these tables that there are some minor differences in the results for period and angle data. For $N_* = 1000$ the results of the two different variables are very similar. For $N_* = 3000$ the angle data are noticeably easier to fit, and for $N_* = 10000$ the period data seem slightly easier to fit. The general structure of the tables for both of these variables, however, is almost identical.

In each table it is clear that, with the exception of the MA models, models above order two do not fit the data very often, with the best fit of any of these models being less than 20% in all cases. Of the lower order models, AR(1), ARMA(1, 1) and all five MA models have fairly similar results, with MA(4) and MA(5) being just slightly worse. The results improve greatly as the cluster size grows, with almost 80% pass rates for $N_* = 10000$. All of the clusters share the property of uniform fits among AR(1), ARMA(1, 1) and the MA models, with no one model standing out as significantly better than the rest.

If we combine our analysis of the statistics calculated in tables 3.1 and 3.2 with our analysis of the fitting results, we start to question whether low order models are simply easy to fit to any data when our criterion is used. If this was the case, it would probably indicate that the periods and angles of stars do not truly behave like any of these ARMA models. We will examine this possibility further in section 3.6.2, after we compare our results to the results of the rejected stars.

3.6.1 Testing of the rejected stars

To see if the stars being rejected by our classification scheme were truly more difficult to fit to ARMA models, we ran the same tests again, this time using the period data that were rejected by our classification scheme. The results are given in table 3.9.

starsRejected	atLeastOne	Average	failedPort	failedTurn	failedDiff
524	36.8%	4.2	6.9%	57.3%	19.5%
1395	43.0%	4.6	4.9%	45.6%	20.1%
4802	58.1%	6.0	4.7%	35.8%	5.0%

Table 3.9: Statistics for star periods that were rejected, $N = 150$

If we compare this table with table 3.1, we see that the rejected stars are indeed more difficult to fit.

3.6.2 Cross-fitting

Next we decided to look at how many times stars could be successfully fit to an AR(1) model *and* an MA model. In order to represent our results, we created tables with two rows and six columns. The first row of the table simply restated the results for the fitting of the MA models, giving the percentage of stars that were successfully fit to each MA model. The second row of the table showed the percentage of stars that were successfully fit to an AR(1) model in the first column, and the next five columns showed the percentage of stars that were successfully fit to both an AR(1) model and each corresponding MA model.

We will refer to this procedure as cross-fitting. The cross-fitting results for the simulations analyzed in the last section are presented in tables 3.10 through 3.15.

	Only AR(1)	MA(1)	MA(2)	MA(3)	MA(4)	MA(5)
Only MA		77.1	78.4	77.7	62.7	53.7
AR(1)	75.8	71.9	71.0	68.5	54.2	47.0

Table 3.10: Percentage of star periods that can be fit to both an AR(1) model *and* an MA(q) model, with $N_* = 1000$, *starsAnalyzed* $= 435$, $N = 150$

	Only AR(1)	MA(1)	MA(2)	MA(3)	MA(4)	MA(5)
Only MA		43.3	40.9	41.7	37.4	35.5
AR(1)	36.0	34.0	32.1	31.8	28.2	26.6

Table 3.11: Percentage of star angles that can be fit to both an AR(1) model *and* an MA(q) model, with $N_* = 1000$, $starsAnalyzed = 650$, $N = 150$

	Only AR(1)	MA(1)	MA(2)	MA(3)	MA(4)	MA(5)
Only MA		47.0	49.7	50.3	42.7	37.0
AR(1)	46.7	38.4	39.3	37.0	30.0	26.6

Table 3.12: Percentage of star periods that can be fit to both an AR(1) model *and* an MA(q) model, with $N_* = 3000$, $starsAnalyzed = 1528$, $N = 150$

	Only AR(1)	MA(1)	MA(2)	MA(3)	MA(4)	MA(5)
Only MA		61.6	61.1	60.3	52.8	49.3
AR(1)	59.0	55.6	54.7	52.4	45.1	41.5

Table 3.13: Percentage of star angles that can be fit to both an AR(1) model *and* an MA(q) model, with $N_* = 3000$, $starsAnalyzed = 2165$, $N = 150$

	Only AR(1)	MA(1)	MA(2)	MA(3)	MA(4)	MA(5)
Only MA		77.1	78.4	77.7	62.7	53.7
AR(1)	75.8	71.9	71.0	68.5	54.2	47.0

Table 3.14: Percentage of star periods that can be fit to both an AR(1) model *and* an MA(q) model, with $N_* = 10000$, $starsAnalyzed = 5165$, $N = 150$

	Only AR(1)	MA(1)	MA(2)	MA(3)	MA(4)	MA(5)
Only MA		71.4	70.9	70.9	62.4	56.7
AR(1)	70.5	66.4	65.5	63.9	55.4	50.0

Table 3.15: Percentage of star angles that can be fit to both an AR(1) model *and* an MA(q) model, with $N_* = 10000$, $starsAnalyzed = 7531$, $N = 150$

From examination of these tables, we can easily see that AR(1) and MA(1) through MA(5) tend to be successfully fit to the same stars, since for each entry in the second row the percentage of times the data was successfully fit to both an AR(1) and an MA(q) model is only slightly less than $\min(\text{AR}(1), \text{MA}(q))$. Thus this suggests that no particular ARMA model is a great fit to the data, but rather it is relatively easy to fit AR(1), ARMA(1, 1) and the first five MA models to a certain percentage of the data.

3.7 Results of extended runs, $N = 250$

At this point we decided to extend the simulations, adding another 100 data points on to the end of each time series and re-running the script on each cluster. We first present the statistics that were calculated by the script. Statistics for period data are given in table 3.16, and the statistics for angle data are given in table 3.17.

N_*	starsLeft	starsRejected	starsAnalyzed	atLeastOne	Average	failedPort	failedTurn	failedDiff
1000	55	470	475	37.7%	4.8	1.3%	55.4%	30.9%
3000	117	1172	1711	52.5%	5.2	3.3%	23.0%	34.1%
10000	51	4229	5720	83.5%	7.3	1.0%	13.1%	4.4%

Table 3.16: Statistics for star periods, $N = 250$

N_*	starsLeft	starsRejected	starsAnalyzed	atLeastOne	Average	failedPort	failedTurn	failedDiff
1000	55	294	651	40.9%	6.0	1.2%	54.7%	22.1%
3000	117	817	2066	63.5%	6.7	1.0%	33.7%	13.9%
10000	51	2589	7360	73.8%	7.2	0.6%	23.3%	8.9%

Table 3.17: Statistics for star angles, $N = 250$

These statistics are very similar to the those presented in tables 3.1 and 3.2, with one important distinction. For each cluster, the percentage of stars that were successfully fit to at least one model is between 8% and 25% lower than the corresponding statistic in the $N = 150$ case. Clearly it becomes more difficult to fit these ARMA models to the data as the series length becomes longer.

The results of the model fitting for these time series are shown in tables 3.18 through 3.23.

	q=0	q=1	q=2	q=3	q=4	q=5
p=0		20.0	18.7	20.4	17.7	17.7
p=1	14.9	18.7	21.4	4.4	1.1	0.6
p=2	8.0	9.9	4.6	0.2	0.0	0.2
p=3	4.2	3.6	0.0	0.0	0.0	0.0
p=4	1.3	0.8	0.0	0.0	0.0	0.0
p=5	1.3	0.2	0.2	0.0	0.0	0.0

Table 3.18: Percentage of star periods that can be fit to ARMA(p,q) models, with $N_* = 1000$, $starsAnalyzed = 475$, $N = 250$

	q=0	q=1	q=2	q=3	q=4	q=5
p=0		28.0	26.6	27.5	24.4	22.9
p=1	24.0	24.7	18.1	9.0	3.8	1.1
p=2	7.7	7.7	3.7	0.2	0.3	0.6
p=3	5.5	2.9	0.5	0.3	0.3	0.0
p=4	4.1	0.9	0.2	0.0	0.0	0.0
p=5	0.6	0.3	0.2	0.2	0.0	0.0

Table 3.19: Percentage of star angles that can be fit to ARMA(p,q) models, with $N_* = 1000$, $starsAnalyzed = 651$, $N = 250$

	q=0	q=1	q=2	q=3	q=4	q=5
p=0		26.5	27.5	27.6	21.5	20.7
p=1	27.0	27.9	21.6	9.6	2.9	1.4
p=2	12.6	11.9	5.7	1.2	0.6	0.5
p=3	8.8	6.6	0.8	0.1	0.1	0.1
p=4	4.8	2.0	0.2	0.1	0.1	0.1
p=5	2.7	0.5	0.2	0.0	0.0	0.0

Table 3.20: Percentage of star periods that can be fit to ARMA(p,q) models, with $N_* = 3000$, $starsAnalyzed = 1711$, $N = 250$

	q=0	q=1	q=2	q=3	q=4	q=5
p=0		51.7	48.5	48.4	41.7	38.0
p=1	48.7	46.5	32.9	13.9	3.7	0.6
p=2	13.8	13.9	5.4	0.4	0.2	0.1
p=3	8.1	4.4	0.2	0.0	0.0	0.1
p=4	4.2	0.4	0.1	0.1	0.0	0.2
p=5	1.2	0.3	0.2	0.1	0.1	0.1

Table 3.21: Percentage of star angles that can be fit to ARMA(p,q) models, with $N_* = 3000$, $starsAnalyzed = 2066$, $N = 250$

	q=0	q=1	q=2	q=3	q=4	q=5
p=0		68.7	69.0	68.2	52.9	45.6
p=1	66.6	63.7	41.0	16.2	4.1	1.4
p=2	28.3	25.8	11.4	1.2	0.7	0.5
p=3	16.9	8.6	0.8	0.2	0.2	0.2
p=4	8.7	1.4	0.1	0.0	0.1	0.1
p=5	2.1	0.3	0.0	0.0	0.0	0.1

Table 3.22: Percentage of star periods that can be fit to ARMA(p, q) models, with $N_* = 10000$, $starsAnalyzed = 5720$, $N = 250$

	q=0	q=1	q=2	q=3	q=4	q=5
p=0		63.6	62.0	62.0	51.7	46.8
p=1	62.5	60.7	40.0	15.2	3.5	0.6
p=2	18.8	17.9	7.1	0.5	0.2	0.2
p=3	10.2	5.2	0.3	0.1	0.1	0.1
p=4	4.7	0.4	0.0	0.0	0.0	0.0
p=5	1.0	0.1	0.0	0.0	0.0	0.0

Table 3.23: Percentage of star angles that can be fit to ARMA(p, q) models, with $N_* = 10000$, $starsAnalyzed = 7360$, $N = 250$

These results are strikingly similar to the results for $N = 150$, with the clear difference being, as expected, a decrease across the board in the percentage of successfull model fits.

3.8 Results for $N = 1000$

The next sensible step to take at this point was to try analyzing even longer time series, to see if the models would eventually stop fitting the data all together. In order to do this we decided to use the same time series as above, but increase the resolution by four times so that the series would now be of length $N = 1000$, with timesteps one quarter of their previous size. This would show us if the percentage fits for each ARMA model continue to drop as the series length increases. We also note that we used series of length $N = 1000$ in section 2.7.4, where we saw that the residual tests perform marginally better for $N = 1000$ than they do for $N = 150$ or $N = 250$.

We present the statistics calculated by the script for these time series in tables 3.24 and 3.25.

N_*	starsLeft	starsRejected	starsAnalyzed	atLeastOne	Average	failedPort	failedTurn	failedDiff
1000	69	359	572	0%	0	10.1%	99.1%	98.6%
3000	135	900	1965	5.1 E-4 %	9	10.4%	99.6%	99.2%
10000	56	3275	6669	1.3 E-3 %	1.3	12.5%	99.3%	92.8%

Table 3.24: Statistics for star periods, $N = 1000$

We can see from these tables that the ARMA models no longer fit the data *at all*! While this seems to show that these ARMA models simply cannot be fit to this data, there are a few burning questions that remain. Why is it that AR(1), ARMA(1, 1) and the first five MA models can be fit to the data 40% to 50% of the time for $N_* = 1000$ and almost 80% of the time for $N_* = 10000$? Are these models simply easy to fit to

N_*	starsLeft	starsRejected	starsAnalyzed	atLeastOne	Average	failedPort	failedTurn	failedDiff
1000	69	201	730	1.4 E-1%	5	14.8%	99.3%	94.4%
3000	135	528	2337	4.3 E-4%	1	12.4%	99.6%	90.9%
10000	56	1734	8210	1.9 E-1%	1.1	12.2%	99.5%	84.1%

Table 3.25: Statistics for star angles, $N = 1000$

random data at series lengths of $N = 150$, and if so, how long a time series do we need to use before we get correct results? Do the fitting results have anything to do with the resolution of the data?

To investigate the effect of the resolution of the data on our fitting procedure, we extended our runs from sections 3.6 and 3.7 so that we would have time series of length $N = 1000$ with the same resolution as our $N = 150$ and $N = 250$ runs. The result was the same as above — the ARMA models could no longer be fit to the data. We also decided to examine the first 150 points of the time series with increased resolution. To our surprise, no models were successfully fit. The discrepancy in the results for time series of length $N = 150$ at different resolutions indicated that there could be something special about the resolution that was used to obtain the results in sections 3.6 and 3.7.

3.9 Testing for trends in the data

At this point we decided to try applying various differencing transformations to the data. The results of the last chapter suggested that applying a log transformation and then differencing the data once was enough to render the data stationary. This conclusion is based on the fact that these transformations were necessary and sufficient to render all nine of the stars we individually analyzed stationary.

In order to test how often it was necessary to difference the data in order to achieve stationarity, we used the same algorithm that we used to classify the stars, except this time we looked at the mean of the data and not the standard deviation. Thus, we calculated the mean of ten individual segments of the data, took the mean of these means, and then compared this value to the mean of each individual segment. If the mean of any of the individual segments differed significantly from the mean of the means, we differenced the data once and repeated the procedure. We used normally distributed noise, which is a stationary time series, as a comparison point to determine how we defined "significantly".

When this test was applied to the time series of length $N = 150$ analyzed in section 3.6, we discovered that each star required exactly one differencing, exactly as predicted. Never the less, we tried differencing all of the stars twice and running them through the script again. This resulted in none of the stars being successfully fit. Three differencings were also tried, with the same result. The same tests were done on the extended $N = 1000$ data sets of the same resolution, and the result was the same. Thus differencing the data once, and only once, produced the best fitting results for $N = 150$, and made no difference for $N = 1000$.

When this test was applied to the more resolved time series analyzed in the previous section, the majority of the stars required only one differencing. However we found that for the $N_* = 1000$ cluster there were 83 stars which needed to be differenced more than once, a number of these required that they were differenced three times. We ran the script on this data differencing the stars twice, and again differencing them three times. In each case the results were the same as before — at this level of resolution, none of the stars were successfully fit to any ARMA models for both $N = 150$ and $N = 1000$.

Seasonal differencing was also attempted on the more resolved data set, to see if the apparent correlation of every fourth point was preventing the data from being successfully fit. Using a seasonal period of four the data was seasonally differencing once, and then twice, but the script was unable to successfully fit any ARMA models to either of these tests.

It appears from the results of this section that the fitting problems were not the result of trends in the data.

3.10 Results of the script run on known models

In order to better interpret our results for different time series lengths, and to determine the overall level of effectiveness of our procedure, we decided to see how our script would behave given instances of actual ARMA models as input data. The results of these test cases would give us a better idea as to how well the script identifies an ARMA model when we know that the model fits the data, and these results would also indicate the number of false positives we could expect at different time series lengths.

To implement our test cases, we used the same seven ARMA models that were used in evaluating the performance of the residual tests in section 2.7.4. We generated the test cases using time series of length 150 and 1000, and using 500 and 5000 instances of each model. We will present the results of the AR$(0.12, 0.1, 0.34)$ and ARMA$(\{0.17, 0.22, 0.41\}, \{0.15, 0.23\})$ test cases, as their results are quite instructive. The statistics computed when the script was run on the data generated using these two models are given in tables 3.26 and 3.27. Note that we no longer include the statistics $starsLeft$, $starsRejected$ or $starsAnalyzed$, for obvious reasons.

N	N_*	atLeastOne	Average	failedPort	failedTurn	failedDiff
150	500	100%	9.1	0%	0%	0%
1000	500	99.6%	9.2	4.0 E-1 %	0%	0%
150	5000	99.8%	9.1	2.4 E-1 %	0%	0%
1000	5000	99.8%	9.2	2.4 E-1 %	0%	0%

Table 3.26: Statistics for the AR(3) model

N	N_*	atLeastOne	Average	failedPort	failedTurn	failedDiff
150	500	99.6%	9.1	4.0 E-1 %	0%	0%
1000	500	99.8%	9.6	2.0 E-1 %	0%	0%
150	5000	99.8%	9.1	1.6 E-1 %	0%	0%
1000	5000	99.6%	9.5	3.2 E-1 %	0%	0%

Table 3.27: Statistics for the ARMA(3, 2) model

When we examine the statistics for $N = 150$, we immediately notice that they are quite different from tables 3.1 and 3.2. Almost all of the test cases can be fit to at least one model, and the average number of fits for each of these stars is greater than nine. Considering that we are examining ten ARMA models in each case, we see that almost every single ARMA model is successfully fit to the data! It is also interesting to note that the results do not vary much when the number of instances increases from 500 to 5000, nor do they vary greatly with series length (though the average number of successful fits goes up slightly).

Before we draw any more conclusions from these statistics we will present the model fitting results for these test cases. These are given in tables 3.28 through 3.35.

	q=0	q=1	q=2	q=3	q=4	q=5
p=0		97.2	95.4	91	45	9.8
p=1	58.6	94.6	87.2	36.8	3.6	0.6
p=2	84.6	87.6	36.4	1.8	0.4	0
p=3	48.6	23.8	1.4	0.4	0.0	0.0
p=4	22.0	4.6	1.2	0.0	0.0	0.0
p=5	3.2	0.2	0.0	0.0	0.0	0.0

Table 3.28: Percentage of instances of the AR(3) model that can be fit to ARMA(p, q) models, with $N_* = 500$, $N = 150$

	q=0	q=1	q=2	q=3	q=4	q=5
p=0		0.0	0.2	2.4	1.6	2.4
p=1	3.2	45.8	63.6	68.4	51.8	21.8
p=2	61.0	65.0	37.8	43.0	14.6	3.6
p=3	86.4	78.2	66.8	19.6	2.8	0.4
p=4	60.8	51.2	18.2	1.6	0.2	0.0
p=5	20.2	9.0	3.2	0.2	0.0	0.0

Table 3.29: Percentage of instances of the ARMA$(3, 2)$ model that can be fit to ARMA(p, q) models, with $N_* = 500$, $N = 150$

	q=0	q=1	q=2	q=3	q=4	q=5
p=0	0.0	0.0	0.0	11.2	8.8	5.6
p=1	0.0	0.0	0.0	48.2	76.2	51.2
p=2	0.0	0.0	0.0	67.0	56.6	8.2
p=3	98.8	98.8	98.4	94.6	70.8	28.4
p=4	27.2	26.6	13.0	2.6	0.0	0.0
p=5	15.4	11.0	2.4	0.0	0.0	0.0

Table 3.30: Percentage of instances of the AR(3) model that can be fit to ARMA(p, q) models, with $N_* = 500$, $N = 1000$

	q=0	q=1	q=2	q=3	q=4	q=5
p=0		0.0	0.0	0.0	0.0	0.0
p=1	0.0	0.0	5.2	5.0	50.4	54.6
p=2	0.0	0.0	2.0	16.4	44.8	6.2
p=3	53.6	73.8	98.6	91.8	29.4	2.6
p=4	97.6	94.4	89.6	20.2	0.8	0.0
p=5	55.6	44.8	17.4	0.6	0.0	0.0

Table 3.31: Percentage of instances of the ARMA$(3, 2)$ model that can be fit to ARMA(p, q) models, with $N_* = 500$, $N = 1000$

	q=0	q=1	q=2	q=3	q=4	q=5
p=0		26.1	14.1	76.3	63.2	30.9
p=1	37.9	39.7	21.1	85.1	42.1	10.8
p=2	33.0	22.4	8.0	44.1	7.84	1.1
p=3	96.9	92.4	64.0	25.5	4.9	1.0
p=4	22.4	14.5	4.1	0.6	0.2	0.1
p=5	10.1	4.5	0.9	0.1	0.0	0.0

Table 3.32: Percentage of instances of the AR(3) model that can be fit to ARMA(p, q) models, with $N_* = 5000$, $N = 150$

	q=0	q=1	q=2	q=3	q=4	q=5
p=0		0.0	0.2	2.9	1.7	2.1
p=1	2.6	42.0	64.7	68.4	51.0	22.6
p=2	63.8	64.4	39.8	41.5	13.2	2.0
p=3	85.4	76.7	65.2	22.0	3.8	0.6
p=4	63.8	52.2	19.8	3.0	0.4	0.1
p=5	20.2	8.4	1.7	0.2	0.0	0.0

Table 3.33: Percentage of instances of the ARMA$(3, 2)$ model that can be fit to ARMA(p, q) models, with $N_* = 5000$, $N = 150$

	q=0	q=1	q=2	q=3	q=4	q=5
p=0		0.0	0.0	10.7	9.6	5.0
p=1	0.0	0.0	0.0	49.9	73.8	49.4
p=2	0.0	0.0	0.0	68.6	56.9	8.8
p=3	99.0	98.9	98.6	95.0	70.4	26.1
p=4	25.5	24.4	16.6	1.9	0.1	0.0
p=5	15.4	12.0	2.5	0.0	0.0	0.0

Table 3.34: Percentage of instances of the AR(3) model that can be fit to ARMA(p, q) models, with $N_* = 5000$, $N = 1000$

	q=0	q=1	q=2	q=3	q=4	q=5
p=0		0.0	0.0	0.0	0.0	0.0
p=1	0.0	0.0	4.3	4.3	52.2	55.6
p=2	0.0	0.4	0.9	17.4	45.3	6.8
p=3	54.6	75.6	98.5	90.7	24.2	1.7
p=4	97.2	96.1	89.4	19.1	0.9	0.1
p=5	54.9	43.6	14.2	0.4	0.0	0.0

Table 3.35: Percentage of instances of the ARMA$(3, 2)$ model that can be fit to ARMA(p, q) models, with $N_* = 5000$, $N = 1000$

We can see that these tables have one extremely useful property — for $N = 1000$ models of order lower than the model of the test case and *all five* of the MA models are almost never fit to the data! When we consider the fact that our statistics presented in tables 3.26 and 3.27 show that nearly every fit that is attempted is successful (this is true greater than 90% of the time for all combinations of N_* and N), we can conclude that models of order lower than the order of the test case are not often among the ten best models, as ranked by the Hannan-Rissanen algorithm. This result was also seen in the AR(2) and ARMA(2, 1) test cases. The MA(3) test case was a slight exception to this rule, with MA(2) being fitted successfully 45.7% of the time.

These results have another important feature. For $N = 1000$ the model that was used to generate the data is successfully fitted over 98% of the time! While this feature is also present in some of the time series of length 150, there are notable exceptions, and one of them is the ARMA(3, 2) model present here, passing only 65% of the time when it was used to generate the test data. To see how often this feature was present in the rest of the test cases, we present the percentage of times the model that was used to generate the test case was correctly identified by our script for the two different series lengths and all seven ARMA test cases in tables 3.36 and 3.37.

$\mathbf{N}_*$	AR(1)	AR(2)	AR(3)	MA(1)	MA(3)	ARMA(2, 1)	ARMA(3, 2)
500	97.6	99.2	97.2	95.4	80.6	86.6	78.2
5000	97.2	98.0	96.7	94.7	80.7	86.1	65.2

Table 3.36: Percentage of instances of each ARMA test case that were correctly identified, $N = 150$

$\mathbf{N}_*$	AR(1)	AR(2)	AR(3)	MA(1)	MA(3)	ARMA(2, 1)	ARMA(3, 2)
500	99.2	98.8	98.8	97.2	93.8	97.6	98.6
5000	98.4	98.6	99.0	97.9	91.7	97.6	98.5

Table 3.37: Percentage of instances of each ARMA test case that were correctly identified, $N = 1000$

As we can see, for $N = 150$ the AR models and the MA(1) model were identified over 95% of the time, while the MA(3) model and the two ARMA models were not as successful. This is probably due to the fact that the higher order models have greater values of the BIC statistic, and thus they are sometimes not among the ten models that the Hannan-Rissanen algorithm choses. Clearly this problem is eliminated when a series length of $N = 1000$ is used.

Using the results of these tests we can draw some conclusions about the behaviour of our script. We saw in section 2.7.4 that a series length of $N = 150$ is sufficient to assure that the residual tests behave as expected. However, time series of length $N = 150$ are not long enough for us to obtain accurate fitting data from our script. With a series length of $N = 1000$, there will be a block of very small percentages above and to the left of an ARMA model which can be successfully fit to the data. Also, if the model is not an MA model then no MA models will be identified, and the correct model should be identified over 95% of the time. The three AR test cases were correctly identified over 97% of the time using a time series length of only $N = 150$, however the results for the MA and other ARMA models were not very successfull at this length. It is also clear that what $N = 150$, lower order models will often be incorrectly identified.

At this point the question of whether or not multiple different instances of a particular model would always have exactly the same fitting results was worth asking. If this was the case, then given that enough instances of a particular model were generated and fitted, the percentage of each AR, MAand ARMA model that was successfully fit would tell us something about the properties of that model, and we would see the same set of results whenever a set of data was a good fit to a specific model.

This idea was tested by repeatedly running the script on the same test case, and it was indeed the case that for a particular model the percentage of each other model that could be fit to it was the same. However, this was only true if the parameters were identical — an AR(0.3) model did not have the same fit table as an AR(0.8) model. Thus any attempt to compare test results directly with data would require generation of endless test cases.

3.11 Discussion and conclusions

3.11.1 Results of the ARMA analysis

Given the results of the test cases applied to our script, combined with the fact that when time series of length $N = 1000$ were examined neither the period nor the angle data of stars could be successfully fit to the ARMA models we tested, we conclude that these ARMA models cannot be successfully fit to period or angle data from an $N-$body simulation. There are, however, various interesting results and possibilities to consider.

It is clear that the data we studied initially has some kind of correlation, evidence of this being that when a different resolution was used, the fitting percentages all dropped to zero. We also note that our test cases show that models which are incorrectly fit to the data maintain approximately the same percentage of fits for different values of N_*. We saw in section 3.6 that as we increased cluster size, the fit percentage for all of the models increased drastically, with AR(2) reaching 78.4% for period data with $N_* = 10000$. Clearly this indicates that whatever is causing these results, it becomes more pronounced as the cluster size increases. Discovering what causes this correlation could yet provide interesting information about the dynamics of globular clusters.

3.11.2 Effectiveness of an automated ARMA analysis on large sets of data

The results of the test cases we examined in section 3.10 indicate that this approach to ARMA analysis is very promising. If series lengths of $N = 1000$ and at least 500 instances of the data are used, a model that fits the data well is easy to determine using our procedure — it should be successfully fitted about 95% of the time, and models of lower order will be successfully fitted very rarely, leaving a block of very small percentages in the fit tables above and to the left of the appropriate model. Detection of higher order models should also be possible, and this idea is described below.

3.12 Future work

3.12.1 Higher order models

While we have shown that ARMA models of order five or less cannot be successfully fit to period or angle data from an $N-$body simulation, this does *not* mean that ARMA models of a higher order do not work. Testing higher order models would involve increasing the number of ARMA models examined, as well as the total number of models cleared for testing by the Hannan-Rissanen algorithm. In order to perform such an analysis, we would realistically require that the entire procedure was programmed in a language such as C, as doing such an analysis in Mathematica® would require weeks or possibly even months, depending on how many models were analyzed. Also, the number of models chosen by the Hannan-Rissanen procedure would have to be carefully selected. If too many models were selected for testing, then models of order lower than the correct model could start to appear in the fitting tables more frequently. If too few models were selected, then it is possible that models that fit the data would not always be chosen, and we would get less of a spike than we saw in our test cases above.

3.12.2 Heteroskedastic modelling

The classification procedure we described in section 3.2 was designed to minimize the number of stars with non-constant variances being analyzed. It is still possible, however, that the variance of the stars which were accepted by our scheme was large enough to prevent us from modelling the data with an ARMA model. If this was the case, using models that account for heterskedasticity such as Autoregressive Conditional Heteroskedasticity (ARCH), Generalized Autoregressive Conditional Heteroskedasticity (GARCH) or Generalized Autoregressive Conditional Heteroskedasticity Moving Average (GARCHMA) models could be more successful.

Considering the fact that designing an automated procedure to fit data to these models would be a very challenging undertaking, and noting that our classification scheme rejected more period data than it did angle data, further individual analysis of angle data using these models would probably be the most sensible continuation of this work.

Chapter 4

The Holtsmark Distribution

4.1 Introduction

4.1.1 Motivation

The Holtsmark distribution has been unchallenged as the definitive distribution of forces in a globular cluster ever since its derivation by Professor Holtsmark in 1917 [34]. It was re-derived by Chandrasekhar in 1943 [23], and this is still the standard derivation in the literature today. We tested the validity of the Holtsmark distribution as a routine matter, and thus we were quite surprised to discover that it *does not* correctly predict the force distribution for spherical clusters of stars.

As we will see, Chandrasekhar believed that he had proven that the dominant contribution to the force on any star is almost exclusively due to the nearest neighbour to that star, and this was used to justify the assumption that the force distribution of an infinite cluster would be the same as that of a real cluster with a finite radius. This idea and the related idea that the contribution to the force distribution made by distant stars is practically insignificant are usually taken as given in the literature

([36], [40], [15]). Kandrup even goes so far as to declare "Consequently, it would appear that standard treatments significantly overestimate the influence of distant stars. A correct treatment demonstrates an appropriate cut-off for the large distance logarithmic divergence at a distance of the order of the mean separation between stars" [35]. We will see that, while these widely held beliefs are quite true for stars close to the center of the cluster, stars which are located in the outer regions of the cluster are significantly influenced by the far field.

The distribution of forces is a fairly basic property of a globular cluster. Since we discovered that the Holtsmark distribution is not correct, an obvious avenue of interest was to find out what exactly was wrong with it, and how this could be corrected. The force distribution contains valuable information — it could be used, for example, to estimate star loss rates for clusters of various sizes.

Another possibility would be to use the distribution of forces to run an $N-$body simulation. A typical $N-$body algorithm, such as the treecode algorithm described in chapter 1, runs in $O(NlogN)$ time [1]. If the force distribution was to be used to run an $N-$body simulation, this would run in $O(NM)$ time, where M is a variable associated with sampling from the force distribution. Thus this type of simulation would run faster than a standard $N-$body simulation for large values of N. This simulation would be non-physical, since there would be no continuity in the force experienced by a particular star at consecutive time steps. However the large scale properties of such a simulation would likely be similar to that of a standard $N-$body simulation.

[1] We remind the reader at this juncture that throughout this chapter N will be used to signify the number of stars in a cluster

4.1.2 Chapter outline

In this chapter we first give an expanded version of Chandrasekar's derivation of the Holtsmark distribution, which can be found in [23]. Next we describe how this distribution was tested, and we show that it does not fit the real force distribution. We then explain why it is that the distribution is incorrect, and we give the beginnings of a (much more complicated) derivation of the force distribution in a star cluster, one which does not make the flawed assumptions made in the derivation of the Holtsmark distribution. The analytical work relating to the derivation of a generalized force distribution was largely carried out by J. C. Lewis in consultation with myself, and has not up to this point been published or disseminated.

4.2 Derivation of the Holtsmark Distribution

4.2.1 Markov's method

Given a random variable $\boldsymbol{y}$ which is the sum of N independent random variables $\boldsymbol{x}_i$, the probability density for $\boldsymbol{y}$ is given by

$$P_N(\boldsymbol{y}) = \int \delta\left(\boldsymbol{y} - \boldsymbol{x}_1 - \ldots - \boldsymbol{x}_N\right) \prod_{i=1}^{N} p_i\left(\boldsymbol{x}_i\right)\, d^3x_1\, d^3x_2 \ldots d^3x_N. \tag{4.1}$$

Expressing $\delta\left(\boldsymbol{y} - \boldsymbol{x}_1 - \ldots - \boldsymbol{x}_N\right)$ as a Fourier transform we have

$$\begin{aligned} P_N(\boldsymbol{y}) &= \left(\frac{1}{2\pi}\right)^3 \int d^3u\, e^{\imath \boldsymbol{u}\cdot[-\boldsymbol{y}+\boldsymbol{x}_1+\ldots+\boldsymbol{x}_N]} \prod_{i=1}^{N} p_i\left(\boldsymbol{x}_i\right)\, d^3x_1\, d^3x_2 \ldots d^3x_N \\ &= \left(\frac{1}{2\pi}\right)^3 \int e^{-\imath \boldsymbol{u}\cdot\boldsymbol{y}} A_N(\boldsymbol{u})\, d^3u \end{aligned} \tag{4.2}$$

where

$$A_N(\boldsymbol{u}) = \prod_{i=1}^{N} \int e^{\imath \boldsymbol{u}\cdot\boldsymbol{x}_i}\, p_i\left(\boldsymbol{x}_i\right)\, d^3x_i \tag{4.3}$$

is the characteristic function of $p_i(\boldsymbol{x}_i)$, the probability density for the random variable $\boldsymbol{x}_i$. If $\boldsymbol{x}_i$ depends on a parameter ξ_i, which is a random variable with probability density $\varpi(\xi_i)$, then the characteristic function becomes

$$A_N(\boldsymbol{u}) = \prod_{i=1}^{N} \int e^{\iota \boldsymbol{u}\cdot\boldsymbol{x}_i(\xi_i)}\, \varpi_i(\xi_i)\, d\xi_i\,. \tag{4.4}$$

If the random variables have the *same* distribution, then we can remove the i subscripts from eq.(4.4), and the characteristic function becomes

$$A_N(\boldsymbol{u}) = \left[\int e^{\iota \boldsymbol{u}\cdot\boldsymbol{x}(\xi)}\, \varpi(\xi)\, d\xi\right]^N. \tag{4.5}$$

The random variables ξ_i may be of dimension higher than 1, and in fact in the derivation of the Holtsmark distribution they are of dimension 3. Eqs.(4.1 - 4.5) are the principal formulae of Markov's method.

4.2.2 Markov's method applied to inverse–square forces

Suppose that a star is located at the origin of a Cartesian coordinate system. Suppose that it is surrounded by a sphere of radius R containing N stars.

Let $\boldsymbol{x}_i$ be the position of star i in this sphere, and let the acceleration which it imparts to the central star be $\boldsymbol{a}_i$. Clearly

$$\boldsymbol{a}_i = G\,\frac{M_i \boldsymbol{x}_i}{r_i^3} \tag{4.6}$$

with

$$r_i = |\boldsymbol{x}_i| \tag{4.7}$$

and

$$\boldsymbol{a} = \sum_{i=1}^{N} \boldsymbol{a}_i \tag{4.8}$$

is the net acceleration experienced by the central star.

Here we make our first assumption.

Assumption 1: The N stars are independently, identically distributed.

Because of this assumption, we can make use of Markov's method to obtain the characteristic function of $\boldsymbol{a}$. This is given by

$$A_N(\boldsymbol{u}) = \prod_{i=1}^{N} \left[\int_{M_i=0}^{\infty} \int_{r_i=0}^{R} e^{\iota \boldsymbol{u} \cdot \boldsymbol{a}_i} \varpi_i(\boldsymbol{x}_i, M_i)\, d^3x_i dM_i \right], \tag{4.9}$$

where $\varpi_i(\boldsymbol{x}_i, M_i)$ is the probability density of star i.

Another consequence of assumption #1 is that we can now replace $\boldsymbol{x}_i$, r_i, M_i and $\boldsymbol{a}_i$ with $\boldsymbol{x}$, r, M and $\boldsymbol{\alpha}$. We use $\boldsymbol{\alpha}$ to represent the acceleration due to a random star because $\boldsymbol{a}$ is already being used to represent the total acceleration of our test star, as defined in eq.(4.8). Thus we write the characteristic function as

$$A_N(\boldsymbol{u}) = \left[\int_0^{\infty} \int_0^{R} e^{\iota \boldsymbol{u} \cdot \boldsymbol{\alpha}} \varpi(\boldsymbol{x}, M) d^3x dM \right]^N \tag{4.10}$$

where, as explained above,

$$\boldsymbol{\alpha} = \frac{GM\boldsymbol{x}}{r^3}. \tag{4.11}$$

Next if we note that

$$\int_0^{\infty} \int_0^{R} \varpi(\boldsymbol{x}, M) d^3x dM = 1 \tag{4.12}$$

and we add this to and subtract it from the portion of eq.(4.10) within the brackets, we get

$$A_N(\boldsymbol{u}) = \left[1 - \int_0^{\infty} \int_0^{R} \{1 - e^{\iota \boldsymbol{u} \cdot \boldsymbol{\alpha}}\} \varpi(\boldsymbol{x}, M) d^3x dM \right]^N. \tag{4.13}$$

It is at this point that we make our second assumption.

Assumption 2: The cluster can be treated as if the stars are uniformly distributed over an infinite volume.

The assumption of a uniform distribution of stars lets us divide $\varpi(\boldsymbol{x}, M)$ into two parts:

$$\varpi(\boldsymbol{x}, M) = \varpi_{\boldsymbol{x}} \varpi_M . \tag{4.14}$$

We can then write the probability density $\varpi_{\boldsymbol{x}}$ in terms of the constant number density n as

$$\varpi_{\boldsymbol{x}} = \frac{n}{N} = \frac{3}{4\pi R^3}. \tag{4.15}$$

Thus we have

$$\varpi(\boldsymbol{x}\, ,\, M) = \frac{3}{4\pi R^3}\, \varpi_M\, , \tag{4.16}$$

where ϖ_M is the stellar mass distribution.

We implement the assumption of infinite radius by taking the limit as $R \rightarrow \infty$ in such a way as to keep the number density n constant. Combining these two steps we get

$$A(\boldsymbol{u}) = \lim_{R\rightarrow\infty} \left[1 - \frac{3}{4\pi R^3} \int_0^\infty \int_0^R \left\{ 1 - e^{\iota \boldsymbol{u}\cdot\boldsymbol{\alpha}} \right\} d^3x\, \varpi_M dM \right]^{4\pi R^3 n/3} . \tag{4.17}$$

If we then set

$$C(\boldsymbol{u}) = \int_0^\infty \int_0^\infty \left\{ 1 - e^{\iota \boldsymbol{u}\cdot\boldsymbol{\alpha}} \right\} d^3x\, \varpi_M dM, \tag{4.18}$$

we have

$$A(\boldsymbol{u}) = \lim_{R\rightarrow\infty} \left[1 - \frac{3}{4\pi R^3} C(\boldsymbol{u}) \right]^{4\pi R^3 n/3} . \tag{4.19}$$

Taking the Taylor series expansion of this, we note that in the limit of $R \rightarrow \infty$ it is equivalent to the Taylor series expansion of $e^{-nC(\boldsymbol{u})}$, and thus

$$A(\boldsymbol{u}) = e^{-nC(\boldsymbol{u})}. \tag{4.20}$$

We can obtain the probability density $P(\boldsymbol{a})$ from the characteristic function using

$$P\left(\boldsymbol{a}\right)=\frac{1}{8\pi^{3}}\int e^{-\iota\boldsymbol{u}\cdot\boldsymbol{a}}A\left(\boldsymbol{u}\right)\,d^{3}u, \tag{4.21}$$

as was shown earlier.

4.2.3 An integral expression for the Holtsmark distribution

$C(\boldsymbol{u})$ given in eq.(4.18) can be expressed as a pure number. To do this, we start by changing the variable of integration from d^3x to $d^3\boldsymbol{\alpha}$. Recalling that

$$\boldsymbol{\alpha}=\frac{GM\boldsymbol{x}}{r^{3}} \tag{4.22}$$

where $r = |\boldsymbol{x}|$, we have

$$\alpha=|\boldsymbol{\alpha}|=\frac{GM}{r^{2}}. \tag{4.23}$$

Solving for r gives us

$$r=\sqrt{\frac{GM}{\alpha}}. \tag{4.24}$$

Dividing eq.(4.22) by eq.(4.23) we get

$$\frac{\boldsymbol{\alpha}}{\alpha}=\frac{\boldsymbol{x}}{r} \tag{4.25}$$

which we rearrange to obtain

$$\boldsymbol{x}=r\frac{\boldsymbol{\alpha}}{\alpha} \tag{4.26}$$

If we combine this with eq.(4.24) we get

$$\boldsymbol{x}=\sqrt{\frac{GM}{\alpha}}\frac{\boldsymbol{\alpha}}{\alpha}=\left(GM\right)^{1/2}\alpha^{-3/2}\boldsymbol{\alpha}. \tag{4.27}$$

Taking derivatives we arrive at

$$d^{3}x=\frac{1}{2}\left(GM\right)^{3/2}\alpha^{-9/2}\,d^{3}\alpha, \tag{4.28}$$

which is our desired change of variables.

Substituting this change of variables in to eq.(4.18) we have

$$\begin{aligned} C(\boldsymbol{u}) &= \frac{1}{2} G^{3/2} \int_0^\infty \int_0^\infty \{1 - e^{\iota \boldsymbol{u}\cdot\boldsymbol{\alpha}}\} |\boldsymbol{\alpha}|^{-9/2} \, d^3\alpha \, M^{3/2} \varpi_M dM \\ &= \frac{1}{2} G^{3/2} \left\langle M^{3/2} \right\rangle \int_0^\infty \{1 - e^{\iota \boldsymbol{u}\cdot\boldsymbol{\alpha}}\} |\boldsymbol{\alpha}|^{-9/2} \, d^3\alpha. \end{aligned} \tag{4.29}$$

If $\boldsymbol{\alpha}$ is replaced by $-\boldsymbol{\alpha}$ in the above expression the value is unaltered; and if the two expressions are averaged the result is

$$C(\boldsymbol{u}) = \frac{1}{2} G^{3/2} \left\langle M^{3/2} \right\rangle \int_0^\infty \frac{1 - \cos(\boldsymbol{u}\cdot\boldsymbol{\alpha})}{|\boldsymbol{\alpha}|^{9/2}} \, d^3\alpha. \tag{4.30}$$

Integrating this using spherical coordinates with the z axis $\|$ $\boldsymbol{u}$ we obtain

$$C(\boldsymbol{u}) = \frac{1}{2} G^{3/2} \left\langle M^{3/2} \right\rangle \int_0^\infty d\alpha \int_0^\pi d\vartheta \sin\vartheta \int_0^{2\pi} d\varphi \, \frac{\alpha^2}{\alpha^{9/2}} \{1 - \cos(u\alpha\cos\vartheta)\}. \tag{4.31}$$

Setting $\theta = cos(\vartheta)$, changing the variable of integration to θ and doing the integration we get

$$\begin{aligned} C(\boldsymbol{u}) &= \pi \, G^{3/2} \left\langle M^{3/2} \right\rangle \int_0^\infty \frac{d\alpha}{\alpha^{5/2}} \int_{-1}^1 d\theta \{1 - \cos(u\alpha\theta)\} \\ &= \pi \, G^{3/2} \left\langle M^{3/2} \right\rangle u^{3/2} \int_0^\infty \frac{d\alpha}{\alpha^{5/2}} \left\{ 2 - \frac{2}{\alpha} \sin\alpha \right\} \\ &= 2\pi \, G^{3/2} \left\langle M^{3/2} \right\rangle u^{3/2} \int_0^\infty d\alpha \, \frac{\alpha - \sin\alpha}{\alpha^{7/2}}. \end{aligned} \tag{4.32}$$

The integral is obviously a pure number. Less obviously, it is a pure number which can be evaluated in closed form. Integrating successively by parts we have

$$\begin{aligned} \int_0^\infty d\alpha \frac{\alpha - \sin\alpha}{\alpha^{7/2}} &= -\frac{2}{5} \left[\frac{\alpha - \sin\alpha}{\alpha^{5/2}} \right]_0^\infty + \frac{2}{5} \int_0^\infty d\alpha \frac{1 - \cos\alpha}{\alpha^{5/2}} \\ &= -\frac{2}{5} \cdot \frac{2}{3} \left[\frac{1 - \cos\alpha}{\alpha^{3/2}} \right]_0^\infty + \frac{2}{5} \cdot \frac{2}{3} \int_0^\infty d\alpha \, \frac{\sin\alpha}{\alpha^{3/2}} \\ &= -\frac{2}{5} \cdot \frac{2}{3} \cdot 2 \left[\frac{\sin\alpha}{\alpha^{1/2}} \right]_0^\infty + \frac{2}{5} \cdot \frac{2}{3} \cdot 2 \int_0^\infty d\alpha \, \frac{\cos\alpha}{\alpha^{1/2}} \\ &= \frac{8}{15} \int_0^\infty d\alpha \, \frac{\cos\alpha}{\alpha^{1/2}} \\ &= \frac{16}{15} \int_0^\infty d\alpha \, \cos\alpha^2. \end{aligned} \tag{4.33}$$

It can be shown that the value of this integral is

$$\int_0^\infty d\alpha \, \cos\alpha^2 = \frac{\pi^{1/2}}{2^{3/2}}. \tag{4.34}$$

This gives, overall,

$$C\left(\boldsymbol{u}\right) = \frac{4}{15}\left(2\pi G\right)^{3/2}\left\langle M^{3/2}\right\rangle u^{3/2} \tag{4.35}$$

where $u = |\boldsymbol{u}|$ as usual. When eq.(A.1) is substituted into eqs.(4.20, 4.21) the result is

$$P\left(\boldsymbol{a}\right) = \frac{1}{8\pi^3}\int e^{-\iota\boldsymbol{u}\cdot\boldsymbol{a}-\gamma u^{3/2}}\, d^3u \tag{4.36}$$

where

$$\gamma = \frac{4}{15}\left(2\pi G\right)^{3/2}\left\langle M^{3/2}\right\rangle n\,. \tag{4.37}$$

Taking the z axis in the integrand to be $\parallel$ $\boldsymbol{a}$ we obtain

$$\begin{aligned} P\left(\boldsymbol{a}\right) &= \frac{1}{4\pi^2}\int_0^\infty du\, u^2 \int_{-1}^1 d\omega\, e^{-\iota u a\omega-\gamma u^{3/2}} \\ &= \frac{1}{2\pi^2 a^3}\int_0^\infty dx\, x\,\sin x\, e^{-\gamma\left(\frac{x}{a}\right)^{3/2}}\,. \end{aligned} \tag{4.38}$$

This probability density $P\left(\boldsymbol{a}\right)$ is isotropic in $\boldsymbol{a}$ and so we can meaningfully define a probability density for a. It is

$$\mathcal{P}\left(a\right) = 4\pi a^2\, P\left(\boldsymbol{a}\right)\,, \tag{4.39}$$

which gives

$$P(a) = \frac{2}{\pi a}\int_0^\infty x\sin x e^{-\gamma(x/a)^{3/2}}dx. \tag{4.40}$$

A more convenient form of this equation can be obtained by introducing what Chandrasekhar calls a "normal field" [23], defined by

$$Q_H = \gamma^{2/3}. \tag{4.41}$$

Then if we define the change of variables

$$a = \beta Q_H \tag{4.42}$$

eq.(4.40) becomes

$$\begin{aligned} P(a) &= \frac{2}{\pi\beta Q_H}\int_0^\infty x\sin x e^{-\gamma\left(\frac{x}{\beta Q_H}\right)^{3/2}} dx \\ &= \frac{2}{\pi\beta Q_H}\int_0^\infty x\sin x e^{-(x/\beta)^{3/2}} dx \\ &= \frac{H\left(\beta\right)}{Q_H} \end{aligned} \tag{4.43}$$

where $H\left(\beta\right)$ is called the Holtsmark distribution and is given by

$$H\left(\beta\right) = \frac{2}{\pi\beta}\int_0^\infty x\sin x e^{-(x/\beta)^{3/2}} dx. \tag{4.44}$$

In comparing eq.(4.40) and eq.(4.43) we see that while the constant γ is evaluated within the integral in eq.(4.40), we need only divide by Q_H in eq.(4.43), which is much cleaner. Approximations of eq.(4.44) for small and large values of β are given in appendix B.

The simulations that were run for this thesis used $M = 1$ for the mass of each star, so in this case $P\left(F\right) = P\left(a\right)$, and thus eq.(4.43) also gives us the probability density of the force F.

4.3 Calculation of forces

In order to compare the forces that are predicted by the Holtsmark distribution with the actual forces occurring in the simulations, we had to estimate the probability distribution of the forces obtained from the simulations. To do this, we took the accelerations output by the $N-$body simulation and used Matlab® to make histograms showing the number of stars feeling net forces within different ranges, with each range corresponding to a bin in the histogram. When the frequency of each bin was divided by the total number of stars in the simulation, this gave us an estimate of $P(F)$ for that particular timestep (note that the distribution $P(F)$ could change over time).

We note here that although the Holtsmark distribution is not dependent on time, we decided to calculate the force distribution at various points in the simulation, to see if the distribution is indeed constant.

4.4 Calculation and fitting of the Holtsmark distribution

Calculation of the force predicted by the Holtsmark distribution was accomplished using the standard statistical formula [38]

$$P[c \leq a \leq d] = \int_c^d P(a)\, da, \tag{4.45}$$

where $P[c \leq a \leq d]$ is the probability of a having a value between c and d. Since we created histograms using the calculated star forces, we needed to calculate the probability of the acceleration falling within each bin (according to the Holtsmark distribution) by integrating eq.(4.43) from $a = c$ to $a = d$, and then we could compare this value with the corresponding bin of the calculated forces. Substituting eq.(4.43) into eq.(4.45) gives us

$$P[c \leq a \leq d] = \frac{2}{\pi Q_H} \int_c^d \frac{1}{\beta} \int_0^\infty x \sin x e^{-(x/\beta)^{3/2}} dx da. \tag{4.46}$$

From $a = \beta Q_H$ we have

$$da = d\beta Q_H \tag{4.47}$$

and since

$$\beta = \frac{a}{Q_H} \tag{4.48}$$

the limits of integration become

$$\beta = \frac{c}{Q_H}, \beta = \frac{d}{Q_H}. \tag{4.49}$$

Using this change of variables eq.(4.46) becomes

$$\begin{aligned} P[c \leq a \leq d] &= \frac{2Q_H}{\pi Q_H} \int_{c/Q_H}^{d/Q_H} \frac{1}{\beta} \int_0^\infty x \sin x e^{-(x/\beta)^{3/2}} dx d\beta \\ &= \frac{2}{\pi} \int_{c/Q_H}^{d/Q_H} \frac{1}{\beta} \int_0^\infty x \sin x e^{-(x/\beta)^{3/2}} dx d\beta \\ &= \int_{c/Q_h}^{d/Q_H} H(\beta) d\beta. \end{aligned} \tag{4.50}$$

The value of Q_H used in this calculation is given in theory by combining eq.(4.37) and eq.(4.41) to obtain

$$\begin{aligned} Q_H &= \left(\frac{4}{15} (2\pi G)^{3/2} \left\langle M^{3/2} \right\rangle n \right)^{2/3} \\ &= \left(\frac{4}{15} \right)^{2/3} (2\pi G) \left\langle M^{3/2} \right\rangle^{2/3} n^{2/3}. \end{aligned} \tag{4.51}$$

For the simulations we analyzed $G = 1$ and $M = 1$, so Q_H becomes

$$Q_H = \left(\frac{4}{15} \right)^{2/3} (2\pi) n^{2/3}. \tag{4.52}$$

The problem with obtaining the value of Q_H using this equation is that though one can obtain an estimate of the number density n using

$$n = \frac{3N}{4\pi R^3}, \tag{4.53}$$

if the density is not constant throughout the cluster this creates problems. While the Holtsmark distribution may still hold, it is possible that the effective density will be slightly different from this value of n. The King model, which we used to generate the initial conditions of our simulations, has a non-constant density, and thus we needed to find a method of obtaining a more appropriate value of Q_H.

The easiest way to do this was to fit eq.(4.50) to the frequency of each bin of the histogram that we created from the calculated force data. In order to perform this fit we needed to make a reasonable estimate of the value of Q_H. This was done using a guess and check method — first we choose a value of n and used this to calculate Q_H. Then we compared the force distribution given by eq.(4.50) with the calculated forces, and plotting the two histograms we visually determined how n needed to be adjusted to line the distributions up. Once an appropriate value of n was found, we used this to calculated the value of Q_H to be used as an estimate in the fitting procedure.

All of the numerical integrations and fitting calculations required for this work were done using Mathematica®.

4.4.1 Implementation

The default setting of Mathematica®'s NIntegrate function (which performs numerical integration) uses Gauss-Kronrod integration [10], however this method alone does not work for eq.(4.44). Instead the Method → Oscillatory option of NIntegrate must be used, an option designed to handle integrals involving trigonometric functions. The algorithm used when Method → Oscillatory is specified is often called "integration between the zeroes". It finds (some of) the zeros of the oscillatory function, and integrates between them using Gauss-Kronrod integration. It then uses sequence convergence acceleration via NSum to find the approximate value of the integral [11].

For the fitting of Q_H, the Mathematica® function NonlinearFit was used. This function takes as input a set of data, a function to be fit to the data, a list of the variables that are passed to the function, a list of the free parameters to be varied to fit the given function to the data, and an optional estimate of the values of these free parameters. In this case the function was defined as holts[c_, d_, Q_] and it implemented the numerical integration discussed above, integrating eq.(4.50). The data set passed in consisted of the bin values of the histogram made from the calculated forces, and for each bin the limits of the bin were taken as the c and d values. The result that eq.(4.50) was fit to was given by

$$\text{result} = \frac{\#\text{ of stars in the bin}}{\#\text{ of stars in the cluster}}.$$

Thus the data set passed in had the form $\{\{c_1,d_1,result_1\},\{c_2,d_2,result_2\},...,\{c_n,d_n,result_n\}\}$, where n is the number of bins used in the histogram. For the work done for this thesis $n = 80$ bins were used. The complete call to NonlinearFit was

NonlinearFit[data, holts[*c*, *d*, *Q*], {*c*, *d*}, {*Q*, estimateQ}]

where estimateQ was the initial estimate of the value of Q_H.

Though it does eventually give the correct result, when the NonlinearFit function is used in this fashion it takes a very long time to do each fit, and produces endless streams of error messages associated with oscillating functions and recursion. A good way to improve the performance of Mathematica®'s NonlinearFit function is to teach Mathematica® the partial derivatives of the function being fitted [12]. In order to determine the partial derivatives of eq.(4.50) with respect to c, d and Q_H we let

$$f(\beta, c, d, Q_H) = P[c \leq a \leq d] = \int_{\frac{c}{Q_H}}^{\frac{d}{Q_H}} H(\beta)d\beta. \tag{4.54}$$

Then

$$\frac{\partial f}{\partial c} = \frac{\partial}{\partial c}\left[\int H(\beta)d\beta\right]_{\frac{c}{Q_H}}^{\frac{d}{Q_H}} = \frac{\partial}{\partial c}\left[F(\frac{d}{Q_H}) - F(\frac{c}{Q_H})\right] = -\frac{H(\frac{c}{Q_H})}{Q_H} \tag{4.55}$$

where

$$F(\beta) = \int H(\beta)d\beta. \tag{4.56}$$

Similarly we have

$$\frac{\partial f}{\partial d} = \frac{H(\frac{d}{Q_H})}{Q_H} \tag{4.57}$$

and

$$\frac{\partial f}{\partial Q_H} = c\frac{H(\frac{c}{Q_H})}{Q_H^2} - d\frac{H(\frac{d}{Q_H})}{Q_H^2}. \tag{4.58}$$

We taught Mathematica® these derivatives using the commands

```
Derivative[1, 0, 0][holts] = holtsc;
Derivative[0, 1, 0][holts] = holtsd;
Derivative[0, 0, 1][holts] = holtsQ;,
```

where the function holts implements eq.(4.54) and holtsc, holtsd and holtsQ are the partial derivatives defined above. With these derivatives defined the fitting procedure worked quite well, provided that the guess given by estimateQ was fairly close to the best fit Q_H value.

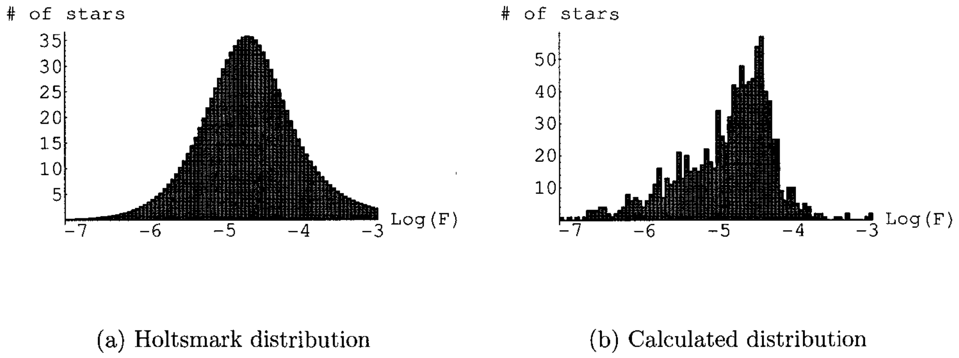
Once the value of Q_H that best fit the holts[c, d, Q] function to the data was found, eq.(4.50) was integrated multiple times over the range of each bin of the histogram of the calculated forces. These values were made into a histogram to be compared with the histogram made using the calculated forces. The results are shown in the next section.

4.5 Results

The initial conditions for the simulations studied were generated using a King model with $\Psi_0 = 5$. Results were obtained for clusters of various sizes at various points in the simulation.

4.5.1 $N_* = 1000$

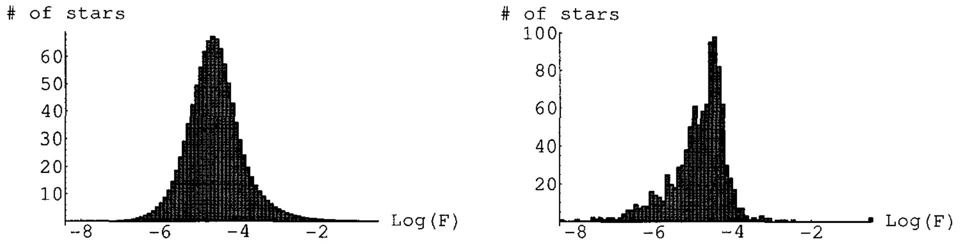
The first tests were conducted using clusters of size $N_* = 1000$. The results are shown below.



(a) Holtsmark distribution

(b) Calculated distribution

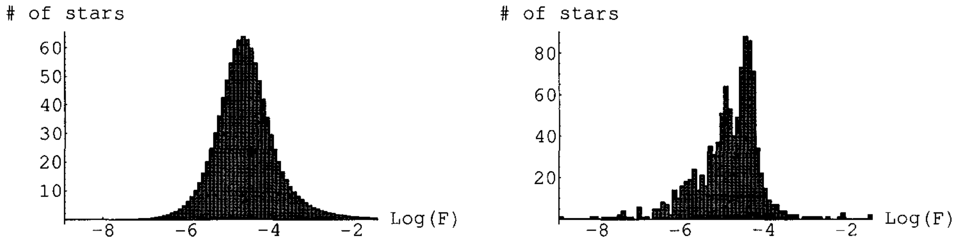
Figure 4.1: Results for $N_* = 1000$, $t = 0$



(a) Holtsmark distribution

(b) Calculated distribution

Figure 4.2: Results for $N_* = 1000$, $t = 500$



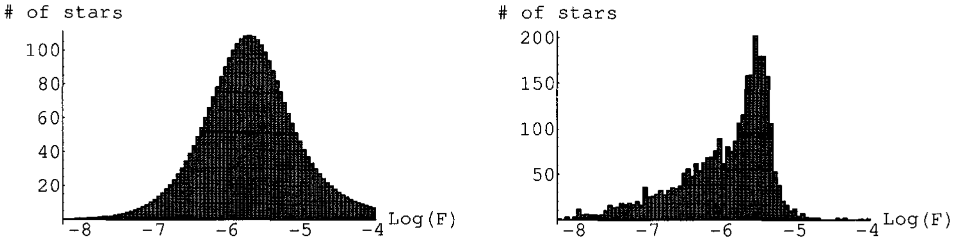
(a) Holtsmark distribution

(b) Calculated distribution

Figure 4.3: Results for $N_* = 1000$, $t = 1000$

4.5.2 $N_* = 3000$

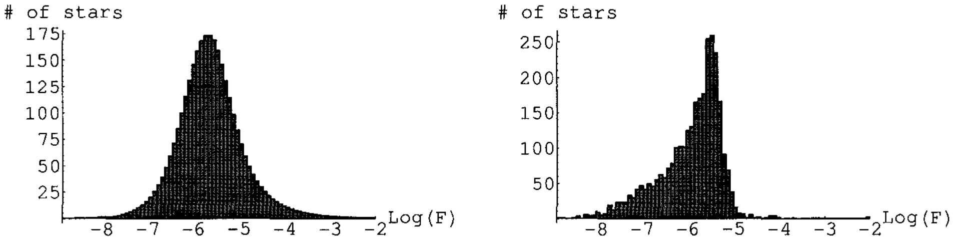
Next clusters of size $N_* = 3000$ were used in the testing, and again the results are shown below.



(a) Holtsmark distribution

(b) Calculated distribution

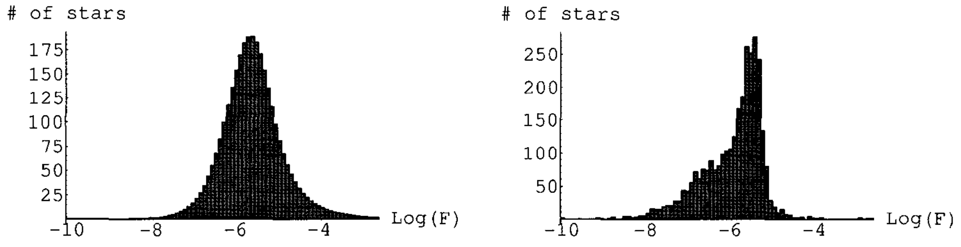
Figure 4.4: Results for $N_* = 3000$, $t = 0$



(a) Holtsmark distribution

(b) Calculated distribution

Figure 4.5: Results for $N_* = 3000$, $t = 500$



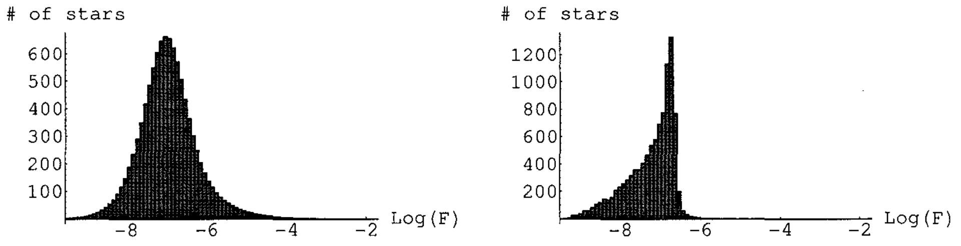
(a) Holtsmark distribution

(b) Calculated distribution

Figure 4.6: Results for $N_* = 3000$, $t = 1000$

4.5.3 $N_* = 10000$

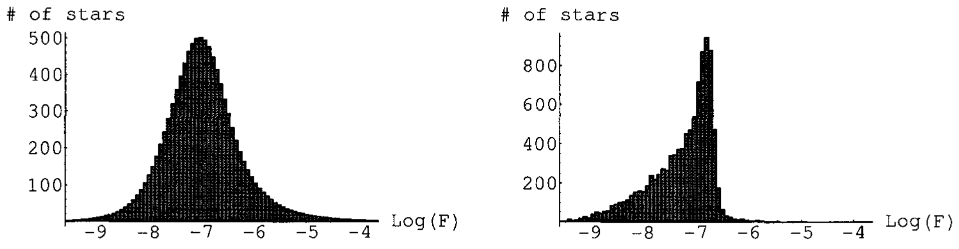
Finally the Holtsmark distribution was tested for a cluster of size $N_* = 10000$. The results are given below.



(a) Holtsmark distribution

(b) Calculated distribution

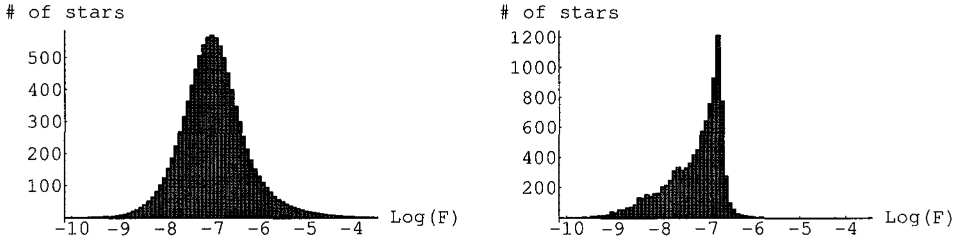
Figure 4.7: Results for $N_* = 10000$, $t = 0$



(a) Holtsmark distribution

(b) Calculated distribution

Figure 4.8: Results for $N_* = 10000$, $t = 500$



(a) Holtsmark distribution

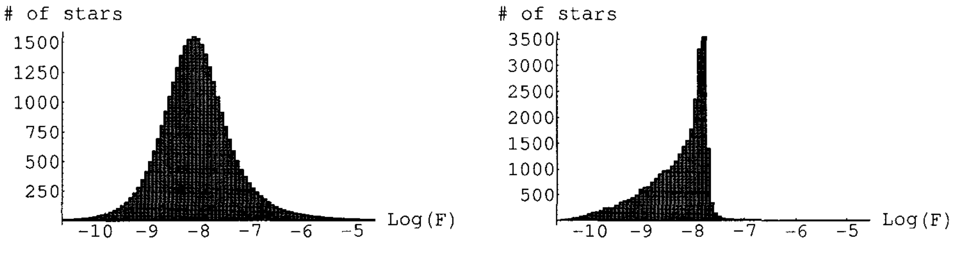
(b) Calculated distribution

Figure 4.9: Results for $N_* = 10000$, $t = 1000$

4.5.4 $N_* = 30000$, 50000 and 100000

When the above results are examined, there is one very noticeable feature. The calculated distribution of forces does not change very much as the simulation progresses — in each case it maintains the same structure. Simulations of clusters bigger than

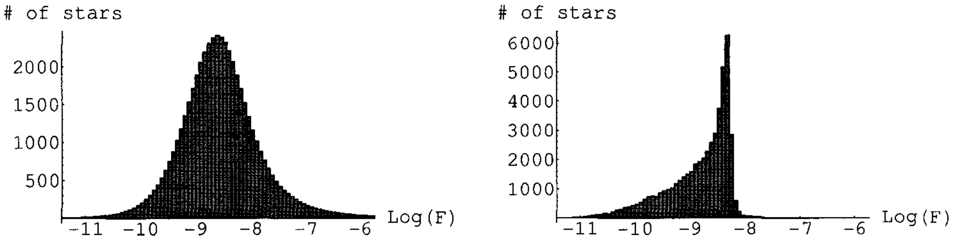
$N_* = 10000$ are very computationally expensive, and given the consistency of the earlier results it was decided that we would obtain results for clusters of size 30000, 50000 and 100000 only at time $t = 0$. The results follow.



(a) Holtsmark distribution

(b) Calculated distribution

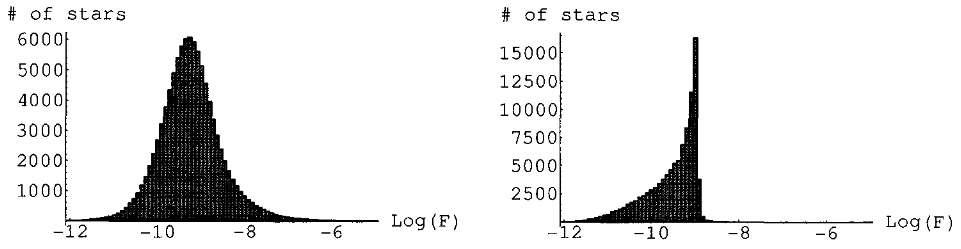
Figure 4.10: Results for $N_* = 30000$, $t = 0$



(a) Holtsmark distribution

(b) Calculated distribution

Figure 4.11: Results for $N_* = 50000$, $t = 0$



(a) Holtsmark distribution

(b) Calculated distribution

Figure 4.12: Results for $N_* = 100000$, $t = 0$

4.6 Discussion of results

It was mentioned above that in all of these results, the calculated distribution stays nearly constant throughout the simulation. We noted earlier that this is the prediction of the Holtsmark distribution, and it appears to be true to a first approximation, at least for the length of the simulations we studied.

All of the calculated force results appear to have a staircase structure on the left side, and then drop off quite quickly on the right. While $N_* = 1000$ is not a big enough cluster for this structure to be clear, it is very evident for $N_* \geqq 10000$. This structure *does not* match that of the Holtsmark distribution, which is almost a symmetric distribution.

A possibility that was considered was that the actual distribution of forces could be the sum of multiple Holtsmark distributions. This was investigated, and it turns out that the Holtsmark distribution maintains its general shape as the value of Q_H is varied, and thus it would impossible to recreate the sharp cutoff that is seen in the data using the sum of multiple Holtsmark distributions.

It was clear at this point that the Holtsmark distribution, or some combination of

Holtsmark distributions, did not match the calculated force distribution. The reason for this difference in structure was initially assumed to be the result of a computational bug, and thus we proceeded to examine this possibility.

4.6.1 Testing of the force calculations

In order to be sure that these results weren't simply the result of a bug in the $N-$body simulation's acceleration output option, we wrote a C program which reads in the star positions, calculates the net force on each star, and then compares this to the acceleration outputted by the treecode. To see how the force calculation is done, consider the equation

$$F_i = -\frac{\partial}{\partial x_i} U\left(r\right). \tag{4.59}$$

We're using Cartesian coordinates, so $i = 1$ to 3, and x_i are the standard Cartesian dimensions. This is equivalent to

$$F_i = -\frac{\partial r}{\partial x_i} U'\left(r\right). \tag{4.60}$$

Now since

$$r = \sqrt{x_1^2 + x_2^2 + x_3^2} \tag{4.61}$$

$\frac{\partial r}{\partial x_i}$ is given by

$$\begin{aligned} \frac{\partial r}{\partial x_i} &= \frac{1}{2}\frac{2x_i}{\sqrt{x_1^2 + x_2^2 + x_3^2}} \\ &= \frac{x_i}{r}. \end{aligned} \tag{4.62}$$

To calculate the second part of eq.(4.60), we note that the gravitational potential is given by

$$U\left(r\right) = -\frac{GMm}{r} \tag{4.63}$$

the derivative of which is

$$U'(r) = \frac{GMm}{r^2}. \tag{4.64}$$

In the simulations that were run $G = 1$, $M = 1$ and $m = 1$ so this becomes

$$U'(r) = \frac{1}{r^2}. \tag{4.65}$$

Therefore we can calculate each component of the force by using

$$F_i = -\frac{x_i}{r^3}. \tag{4.66}$$

This equation was used to calculate the three components of the force on one star due to every other star, and then these were combined to give the net force on that star at that timestep. This was done for every star at every timestep that was analyzed, and then these forces were compared to the accelerations that were calculated and output by the $N-$body simulation. As mentioned above, since all the masses in these simulations were set to one, the forces and accelerations for each star were equal. The percent difference in these values was then calculated and output, and the average percent error for all of the stars was $< 0.1\%$ for all the calculations done for this thesis. This small difference in these values was expected, as the $N-$body simulation that we used makes very small approximations in the force calculations in order to speed up the simulation.

The possibility that the unusual results could be the result of a computational bug other than faulty acceleration data (which was ruled out above) was eliminated through further testing, which is described in the next section.

4.7 Discussion and testing of assumptions

The other possible cause of the difference between the Holtsmark distribution and the actual calculated force distribution was that an assumption was made in the derivation of the Holtsmark distribution that was not valid. There were two major assumptions made, and we discuss each one below.

4.7.1 First assumption

Recall that the first assumption was that the stars were independently, identically distributed. This assumption was also made in generating the initial conditions of the simulations! The stars' initial positions were generated using a King model, and they were each generated independent of each other. Since at the beginning of the simulation the difference in the calculated forces and the Holtsmark distribution was present, this assumption could not be the cause of the problem.

It is worth noting that Chandrasekhar predicted that at higher values of F, the Holtsmark distribution will differ from the actual force distribution [23]. He suggested that this is because binaries star systems will form over time in a globular cluster, and the existence of such systems goes against the assumption of randomly distributed stars. In order for two stars to become bound to each other, they would have to have more potential energy than kinetic energy. So we have [23]

$$\frac{|V|^2}{2} = \frac{G(M_1 + M_2)}{r(|V|)}, \tag{4.67}$$

where $|V|$ is the relative velocity of the two stars. Thus, the critical distance at which two stars would become bound together is

$$r(|V|) = \frac{2G(M_1 + M_2)}{|V|^2}. \tag{4.68}$$

Accordingly, the force at this value of r is

$$|F|_{max} = \frac{GM_2}{r(|V|)^2} \tag{4.69}$$

and if we substitute the mean squared velocity for the squared velocity, we get

$$|F|_{max} = \frac{M_2 \left\langle |V|^2 \right\rangle_{Av}^2}{4G(M_1 + M_2)^2}. \tag{4.70}$$

Chandrasekhar argued that while the Holtsmark distribution will predict some forces greater than eq.(4.70), the actual force distribution will very rapidly approach zero at this point.

For our simulations with $N_* = 1000$, $G = M_1 = M_2 = M = 1$ and $< |V|^2 >^2_{Av} \sim 1$, and thus

$$|F|_{max} \sim 0.1. \tag{4.71}$$

As we can see from figure 4.1, the difference in the calculated force and the Holtsmark distribution happens at values of $|F|$ a couple of orders of magnitude lower than this, and thus this is definitely not the cause of this difference.

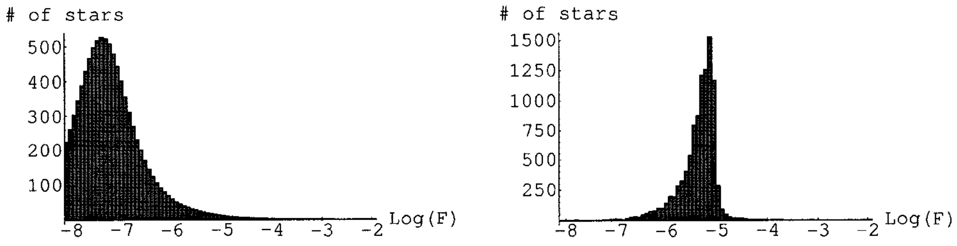
4.7.2 Chandrasekhar's defense of the second assumption

The second assumption was: "The cluster can be treated as if the stars are uniformly distributed over an infinite volume". Chandrasekhar defended this assumption by arguing that the dominant contribution to the force on each star is provided by the nearest neighbour of that star, and thus "the formal extrapolation to infinity of the density of stars obtaining only in a given region of a stellar system can hardly affect the results to any appreciable extent" [23]. He went on to show that, in the limit as F approaches infinity, the force due to the nearest neighbour of a star matches exactly the force predicted by the Holtsmark distribution. He thus concluded that the strongest forces are produced only by the nearest neighbour, and thus assumptions made about the general structure of the cluster should not affect the result of the derivation to any appreciable extent.

While Chandrasekhar's argument may seem convincing at first glance, it is flawed. In Chandrasekhar's defense of this assumption, he assumed that the Holtsmark distribution, derived using this assumption, is correct. He did this when he concluded that the strongest forces are always produced by the nearest neighbour. If an assumption is made in a derivation, the result of that derivation cannot be taken as true in defending the assumption, and this is effectively what he did.

4.7.3 Re-examination of the second assumption

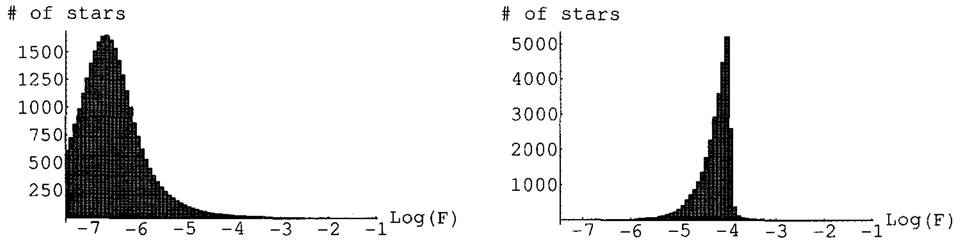
This assumption really has two parts, which are related. In breaking it down, the first question we ask is if the assumption of a uniform distribution of stars is acceptable. In our earlier tests we used clusters that were set up using a King model, and thus the stars were not uniformly distributed. So we decided to test some clusters that *were* set up with a uniform distribution of stars, and we obtained the following results



(a) Holtsmark distribution

(b) Calculated distribution

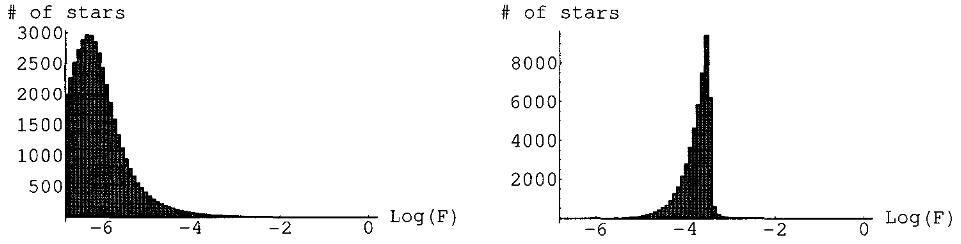
Figure 4.13: Results for $N_* = 10000$, uniform distribution



(a) Holtsmark distribution

(b) Calculated distribution

Figure 4.14: Results for $N_* = 30000$, uniform distribution



(a) Holtsmark distribution

(b) Calculated distribution

Figure 4.15: Results for $N_* = 50000$, uniform distribution

As we can see from these results, the same sharp cutoff exists in the calculated distributions, and thus the assumption of uniform distribution cannot be the cause of the problem. Note that the Holtsmark distribution and the calculated distribution are no longer directly lined up. This is because we did not fit the value of Q_H for the Holtsmark distribution, but instead calculated it using $n = \frac{3N}{4\pi R^3}$, where the position of the farthest star from the center was used as the radius of the cluster. We will discuss the reason for this below.

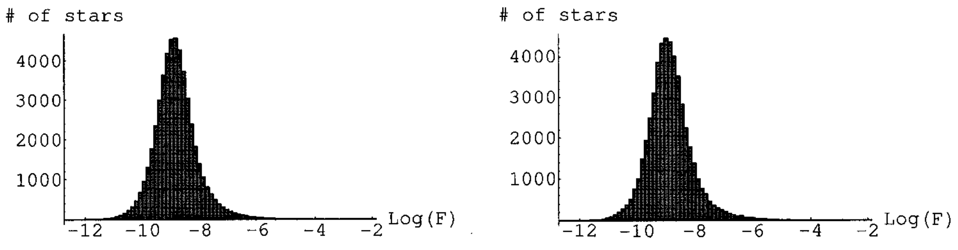
The second part of the assumption says that the cluster can be treated as if it is infinite. This gives us a situation where each star is assumed to be completely surrounded by a uniform number of stars. Clearly this is not the case for stars which are situated in the outer regions of the cluster, as these stars are surrounded by empty space on one side, and stars on the other. Not only is the density lower in the area surrounding these stars, but most of the force on these stars in concentrated towards the center of the cluster. Since there are no balancing forces pulling these stars away from the cluster, we might expect the magnitude of the force acting on these stars to be greater than it would be for stars closer to the middle of the cluster.

If Chandrasekhar's assertion that the vast majority of the force comes from the nearest neighbouring star is correct, then this extra force experienced by stars in the outer regions of the cluster should be insignificantly small. If his assertion is not correct, then it's possible that this extra force causes the actual force distribution to be quite different from the distribution predicted by Holtsmark. Since the cluster being treated as infinite is the only assumption which we have not validated, this is presumably the case.

In order to prove that this assumption is indeed faulty, we designed a test that would approximate a uniform distribution of stars over an infinite volume. Instead of just calculating the force on each star in a cluster, we decided to calculate the force on a test star situated at the center of a cluster. Of course, this only gives us one force. In order to obtain an accurate distribution of forces on this central test star, we decided to generate new uniformly distributed clusters over and over, each time calculating the net force on the center of the cluster.

The only difference between this system and the one that the Holtsmark distribution describes is that the radius does not extend to infinity, something which we can't computationally recreate. However, this *does* remove the problem associated with the varying density on the outer edges of the cluster. The cutoff is symmetric in all directions for this new system, while it is not for the force distribution of all the stars in a single cluster.

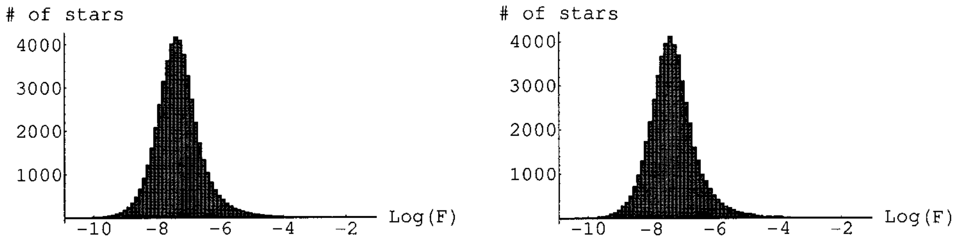
The results of this test are given below for $N_* = 1000$, $N_* = 10000$ and $N_* = 30000$, with $NConfigs = 50000$, where $NConfigs$ is the number of clusters that were generated in each case to obtain the force distribution. This is a sufficiently large value of $NConfigs$ to assure accurate results.



(a) Holtsmark distribution

(b) Calculated distribution

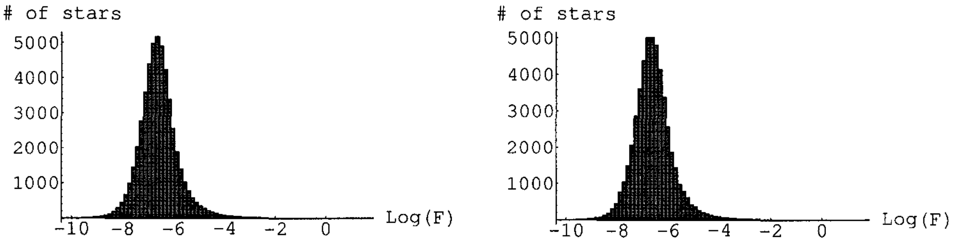
Figure 4.16: Results for $N_* = 1000$, $NConfigs = 50000$



(a) Holtsmark distribution

(b) Calculated distribution

Figure 4.17: Results for $N_* = 10000$, $NConfigs = 50000$



(a) Holtsmark distribution

(b) Calculated distribution

Figure 4.18: Results for $N_* = 30000$, $NConfigs = 50000$

As we can see from these results, the Holtsmark distribution and the calculated distribution are in excellent agreement for this system. Note that again we did not fit Q_H, just like in the uniform distribution test above, because for this system it should be accurately determined by the number density n. As we can see from the agreement in the results above, this seems to be the case.

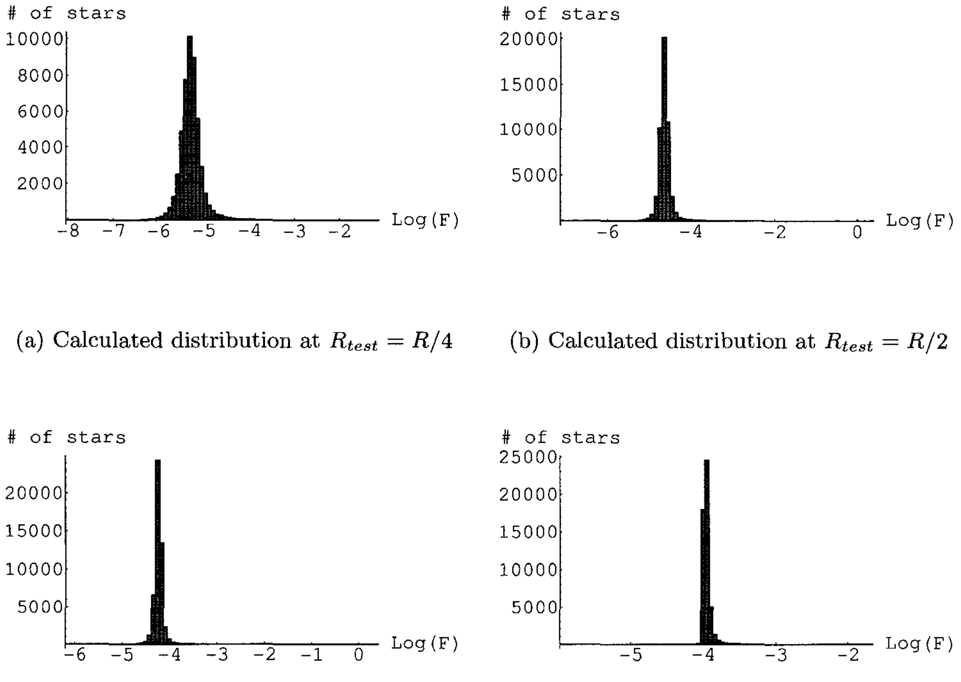
We used the same Matlab® and Mathematica® scripts to generate these results as were used for the initial results. The only difference in the software used was the C programs, and since we throughly tested the C program used in our initial results, we can definitively say that our original results were correct.

We note here that the uniform distribution results shown earlier in figures 4.13, 4.14 and 4.15 show that the calculated force distribution not only doesn't seem to have the same shape as the Holtsmark distribution, but seems to predict much greater forces, on the whole, than the Holtsmark distribution. We can safely say this because the results of our new system line up well with the Holtsmark distribution without fitting of Q_H, and in both cases we have the uniform distribution that Chandrasekhar assumes in his derivation. This is something which could not be determined from examination of the initial results, as the non-uniform distribution meant that Q_H had to be fit. We will discuss the reason for this in the next section.

Clearly we can conclude from the results of this test that the second assumption made in the derivation of the Holtsmark distribution, that being that the cluster can be treated as if the stars are uniformly distributed over an infinite volume, leads to the wrong force distribution. It is clear that the force distribution must depend to some degree on radius. We examine this dependence in the next section.

4.8 Dependence of force distribution on radius

To see the extent of the dependence of the force distribution on radius, we decided to run the multiple configurations test again, this time changing the position of the test star. The results with the test star located at varying fractions of the total radius of the cluster are shown below.



(a) Calculated distribution at $R_{test} = R/4$

(b) Calculated distribution at $R_{test} = R/2$

(c) Calculated distribution at $R_{test} = 3R/4$

(d) Calculated distribution at $R_{test} = R$

Figure 4.19: Results for $N_* = 30000$, $NConfigs = 50000$

We can see from these results that as we increase R_{test}, the lower end of the force distribution F_{min} moves up while F_{max} gets slightly larger. This means that as we move towards the edge of the cluster there are less small forces felt, while a small extra amount of force seems to be added to those stars experiencing the greatest forces. At $R = R_{test}$, the distribution becomes very concentrated at about $F \sim e^{-4}$.

The explanation of this phenomena comes straight from Newton's theorem. If a sphere of radius R and mass M has uniform density, the net gravitational force acting on a test particle located within the sphere at a radius of r is given by

$$F = \frac{GM_r m}{r^2}, \tag{4.72}$$

where M_r is defined as the mass interior to the radius r [29]. Clearly $M_r = 0$ for $r = 0$ and $M_r = M$ for $r \geq R$. Thus, as the test star is moved out towards the edge of the cluster, there is a greater net force from distant stars, pulling it towards the center of the cluster. The effect of this is to add a certain minimal force to the force this test star experiences due to its nearest neighbours.

If the test star is located at the center of the cluster, cases where the closest star is somewhat far away would result in quite a small force (these cases account for the forces of about $F_{min} \sim e^{-8}$ seen in figure 4.18). However as the test star moves farther towards the edge of the cluster, the unbalanced force of stars to one side pulling the test star towards the center assure that the minimum force F_{min} increases, until at the edge of the cluster the majority of the force is provided by background stars, and not the nearest neighbours. This is why the force distribution in figure 4.19d is so concentrated.

It seems that the hypothesis we put forth earlier was correct — stars in the outer regions experience a higher net force, due to the fact that they are not surrounded by stars on all sides. Rather there are many stars to one side of them, pulling them towards the cluster, and there is no balancing force on the other side. We can clearly see that there is a need for a force distribution that takes radius into account. In fact, we will show that this force distribution will also depend on the distribution of mass in the cluster. We discuss the formulation of this more generalized force distribution in the next section.

4.9 Derivation of a generalized force distribution

The derivation of a generalized force distribution starts in much the same way as the derivation of the Holtsmark distribution itself. The important difference in the two lies in the positioning of the test star: the Holtsmark distribution simply places it at the center, which we now know leads to an incorrect force distribution. The more general derivation places the test star at position $\boldsymbol{x}_0$. As we will see, this complicates the situation greatly.

4.9.1 Application of Markov's method

Let $\boldsymbol{x}_i$ be the position of star i and let the acceleration which it imparts to a test star at $\boldsymbol{x}_0$ be denoted by

$$\boldsymbol{a}_i = GM_i \frac{\boldsymbol{x}_i - \boldsymbol{x}_0}{|\boldsymbol{x}_i - \boldsymbol{x}_0|^3}. \tag{4.73}$$

The net acceleration of the test star is given by

$$\boldsymbol{a} = \sum_{i=1}^{N} \boldsymbol{a}_i. \tag{4.74}$$

Again we assume that the stars are independently, identically distributed.

From this point we can follow the same steps as we did in section 4.2.2, up to our second assumption, to obtain

$$A_N(\boldsymbol{u}) = \left[1 - \int \{1 - e^{\iota \boldsymbol{u}\cdot\boldsymbol{\alpha}}\} \varpi(\boldsymbol{x}) d^3x\right]^N \tag{4.75}$$

which is similar to eq.(4.13), with a couple of notable exceptions. First of all, the definition of $\boldsymbol{\alpha}$ has changed to

$$\boldsymbol{\alpha} = GM \frac{\boldsymbol{x} - \boldsymbol{x}_0}{|\boldsymbol{x} - \boldsymbol{x}_0|^3}. \tag{4.76}$$

We also note that while ϖ does depend on M, we have not made that dependence explicit yet (we will later).

Again we use the assumption that the stars are independently distributed to express the probability density ϖ in terms of the number density ρ

$$\varpi_{\boldsymbol{x}} = \frac{\rho(\boldsymbol{x})}{N}. \tag{4.77}$$

This gives us

$$A_N(\boldsymbol{u}) = (1 - \frac{1}{N}C(\boldsymbol{u}))^N \tag{4.78}$$

where

$$C(\boldsymbol{u}) = \int \{1 - e^{\iota \boldsymbol{u}\cdot\boldsymbol{\alpha}}\}\rho(\boldsymbol{x})d^3x. \tag{4.79}$$

In the derivation of the Holtsmark distribution, we took the limit $R \longrightarrow \infty$ of $A_N(\boldsymbol{u})$ and used Taylor series expansions to simplify the expression. While we now know that we cannot do this, taking the limit $N \longrightarrow \infty$ seems quite reasonable, as in practice the value of N is usually very large. Then we have

$$A_N(\boldsymbol{u}) = \lim_{N\longrightarrow\infty}(1 - \frac{1}{N}C(\boldsymbol{u}))^N \tag{4.80}$$

and using the same Taylor series expansion and simplification as before we arrive at

$$A(\boldsymbol{u}) = e^{-C(\boldsymbol{u})}. \tag{4.81}$$

We note that if we set $\rho(\boldsymbol{x}) = n$, where n is the number density in a very large (strictly speaking, infinite) uniform cluster, we obtain

$$C(\boldsymbol{u}) = n\int_0^\infty \{1 - e^{\iota \boldsymbol{u}\cdot\boldsymbol{\alpha}}\}d^3x, \tag{4.82}$$

which makes eq.(4.81) equivalent to eq.(4.20), and thus we have retrieved the Holtsmark distribution (it is trivial to transform to $\boldsymbol{x}_0$ as the origin if $\rho = n$ is constant).

4.9.2 Non-constant density

Again we change our variable of integration from d^3x to $d^3\alpha$. However this time the addition of $\boldsymbol{x}_0$ propagates through, and we end up with

$$\boldsymbol{x} = (GM)^{1/2}\alpha^{-3/2}\boldsymbol{\alpha} + \boldsymbol{x}_0. \tag{4.83}$$

Again taking derivatives gives us

$$d^3x = \frac{1}{2}(GM)^{3/2}\alpha^{-9/2}d^3\alpha. \tag{4.84}$$

Substituting eq.(4.83) and eq.(4.84) into eq.(4.79) we get

$$C(\boldsymbol{u}) = \frac{1}{2}(GM)^{3/2}\int d^3\alpha\,\alpha^{-9/2}\{1-e^{\imath\boldsymbol{u}\cdot\boldsymbol{\alpha}}\}\rho\left[(GM)^{1/2}\alpha^{-3/2}\boldsymbol{\alpha}+\boldsymbol{x}_0\right]. \tag{4.85}$$

Note that the dependence of ρ on M has now been made explicit. One interesting feature of eq.(4.85) is that the averaging over mass becomes a complicated matter, because of this dependence of ρ on M. The mean $\langle M^{3/2}\rangle$ no longer provides a complete characterisation of the effects of mass variation, as it does for the Holtsmark distribution.

We will conclude this chapter with some work that has been done on two different approaches to solving eq.(4.85) for a cluster in which all of the stars have equal mass. In this case we can introduce the reduced variables

$$\boldsymbol{\alpha} = (GM)\,\tilde{\boldsymbol{\alpha}} \tag{4.86}$$

$$\boldsymbol{u} = (GM)^{-1}\,\tilde{\boldsymbol{u}} \tag{4.87}$$

and eq.(4.85) becomes

$$C\left(\tilde{\boldsymbol{u}}\right) = \frac{1}{2}\int d^3\tilde{\alpha}\,\tilde{\alpha}^{-9/2}\left\{1-e^{\imath\tilde{\boldsymbol{u}}\cdot\tilde{\boldsymbol{\alpha}}}\right\}\rho\left[\tilde{\alpha}^{-3/2}\tilde{\boldsymbol{\alpha}}+\boldsymbol{x}_0\right]. \tag{4.88}$$

For our first approach to solving this equation we will integrate using cylindrical coordinates, and in our second approach we will try using spherical coordinates.

4.9.3 Cylindrical coordinate approach

If we combine eq.(4.88) and eq.(4.2) we obtain

$$P\left(\tilde{\boldsymbol{\alpha}}\right) = \left(\frac{GM}{2\pi}\right)^3 \int e^{-\iota\tilde{\boldsymbol{u}}\cdot\tilde{\boldsymbol{\alpha}} - C(\tilde{\boldsymbol{u}})}\, d^3\tilde{u}\,. \tag{4.89}$$

Now using the fact that

$$\begin{aligned}\int_0^{\pi} d\vartheta \sin\vartheta \int_0^{2\pi} d\varphi\, e^{\iota au\cos\vartheta} &= 2\pi\frac{1}{\iota au}\left(e^{\iota au} - e^{-\iota au}\right) \\ &= \frac{4\pi}{au}\sin au \end{aligned} \tag{4.90}$$

we integrate around the $\tilde{\boldsymbol{u}}$ direction as polar axis and get

$$\begin{aligned} P\left(\tilde{\alpha}\right) &= \tilde{\alpha}^2 \int d\vartheta_a d\varphi_a P\left(\tilde{\boldsymbol{\alpha}}\right) \\ &= \tilde{\alpha}^2 \left(\frac{GM}{2\pi}\right)^3 \int \frac{4\pi}{\tilde{\alpha}\tilde{u}} \sin\left(\tilde{\alpha}\tilde{u}\right) e^{-C(\tilde{\boldsymbol{u}})}\, d^3\tilde{u} \\ &= \tilde{\alpha}\frac{(GM)^3}{2\pi^2} \int \frac{\sin\left(\tilde{\alpha}\tilde{u}\right)}{\tilde{u}} e^{-C(\tilde{\boldsymbol{u}})}\, d^3\tilde{u}\,. \end{aligned} \tag{4.91}$$

Next we assume that ρ is spherically symmetric, so that

$$\rho\left[\tilde{\alpha}^{-3/2}\tilde{\boldsymbol{\alpha}} + \boldsymbol{x}_0\right] = \rho\left(\left|\tilde{\alpha}^{-3/2}\tilde{\boldsymbol{\alpha}} + \boldsymbol{x}_0\right|\right) \tag{4.92}$$

and we take the z axis for both integrals to coincide with $\boldsymbol{x}_0$ so that

$$\rho\left(\left|\tilde{\alpha}^{-3/2}\tilde{\boldsymbol{\alpha}} + \boldsymbol{x}_0\right|\right) = \rho\left(\left|\tilde{\alpha}^{-3/2}\tilde{\boldsymbol{\alpha}} + x_0\hat{\mathbf{1}}_z\right|\right) \tag{4.93}$$

where $\hat{\mathbf{1}}_z$ is a unit vector in the z direction and

$$\begin{aligned} \left|\tilde{\alpha}^{-3/2}\tilde{\boldsymbol{\alpha}} + x_0\hat{\mathbf{1}}_z\right|^2 &= \left(\tilde{\alpha}^{-3/2}\tilde{\boldsymbol{\alpha}} + x_0\hat{\mathbf{1}}_z\right)\cdot\left(\tilde{\alpha}^{-3/2}\tilde{\boldsymbol{\alpha}} + x_0\hat{\mathbf{1}}_z\right) \\ &= \tilde{\alpha}^{-3}\tilde{\alpha}^2 + x_0^2 + 2\tilde{\alpha}^{-3/2}\tilde{\boldsymbol{\alpha}}\cdot x_0\hat{\mathbf{1}}_z \\ &= \tilde{\alpha}^{-1} + x_0^2 + \frac{2\alpha_{\|}x_0}{(\alpha_{\|}^2 + \alpha_{\perp}^2)^{3/4}} \\ &= \frac{1}{\sqrt{\alpha_{\|}^2 + \alpha_{\perp}^2}} + \frac{2\alpha_{\|}x_0}{(\alpha_{\|}^2 + \alpha_{\perp}^2)^{3/4}} + x_0^2\,, \end{aligned} \tag{4.94}$$

where $(\alpha_\parallel, \alpha_\perp, \varphi_\alpha)$ is the expression for $\tilde{\boldsymbol{\alpha}}$ in cylindrical coordinates. Similarly, we write $\tilde{\boldsymbol{u}}$ in cylindrical coordinates as

$$\tilde{\boldsymbol{u}} = \left(u_\parallel,\, u_\perp,\, \varphi_u\right) \tag{4.95}$$

whence, from eq. (4.88) we have

$$\begin{aligned} C\left(\tilde{\boldsymbol{u}}\right) = \frac{1}{2} \int \frac{d\alpha_\parallel\, \alpha_\perp d\alpha_\perp\, d\varphi_\alpha}{\left(\alpha_\parallel^2 + \alpha_\perp^2\right)^{9/4}} \left\{1 - e^{\iota\left(u_\parallel \alpha_\parallel + u_\perp \alpha_\perp \cos(\varphi_u - \varphi_\alpha)\right]}\right\} \\ \times \rho \left(\sqrt{\frac{1}{\sqrt{\alpha_\parallel^2 + \alpha_\perp^2}} + \frac{2\alpha_\parallel x_0}{(\alpha_\parallel^2 + \alpha_\perp^2)^{3/4}} + x_0^2} \right) \end{aligned} \tag{4.96}$$

or

$$\begin{aligned} C\left(u_\parallel,\, u_\perp\right) = \pi \int \frac{d\alpha_\parallel\, \alpha_\perp d\alpha_\perp}{\left(\alpha_\parallel^2 + \alpha_\perp^2\right)^{9/4}} \left\{1 - e^{\iota u_\parallel \alpha_\parallel} J_0\left(u_\perp \alpha_\perp\right)\right\} \\ \times \rho \left(\sqrt{\frac{1}{\sqrt{\alpha_\parallel^2 + \alpha_\perp^2}} + \frac{2\alpha_\parallel x_0}{(\alpha_\parallel^2 + \alpha_\perp^2)^{3/4}} + x_0^2} \right) . \end{aligned} \tag{4.97}$$

Then eq. (4.91) becomes

$$P\left(\tilde{a}\right) = \tilde{a} \frac{\left(GM\right)^3}{\pi} \int \frac{\sin\left(\tilde{a}\tilde{u}\right)}{\tilde{u}} e^{-C\left(u_\parallel, u_\perp\right)} u_\perp du_\perp du_\parallel \tag{4.98}$$

$$= \tilde{a} \frac{\left(GM\right)^3}{\pi} \int_0^\infty du_\parallel \int_0^\infty du_\perp u_\perp \frac{\sin\left(\tilde{a}\tilde{u}\right)}{\tilde{u}} \left[e^{-C\left(u_\parallel, u_\perp\right)} + e^{-C\left(-u_\parallel, u_\perp\right)}\right] \tag{4.99}$$

with

$$\tilde{a} = \sqrt{\tilde{a}_\parallel^2 + \tilde{a}_\perp^2} \text{ and } \tilde{u} = \sqrt{\tilde{u}_\parallel^2 + \tilde{u}_\perp^2}\,. \tag{4.100}$$

Hence four integrations are necessary to evaluate P.

A modification

Eqn. (4.97) can be written as

$$\begin{aligned}
\Re\, C\left(u_{\|},\, u_{\perp}\right) =& \frac{\pi}{2}\int \frac{d\alpha_{\|}\,\alpha_{\perp}d\alpha_{\perp}}{\left(\alpha_{\|}^{2}+\alpha_{\perp}^{2}\right)^{9/4}}\left\{1-\cos\left(u_{\|}\alpha_{\|}\right)J_0\left(u_{\perp}\alpha_{\perp}\right)\right\} \\
&\times\left\{\rho\left(\sqrt{\frac{1}{\sqrt{\alpha_{\|}^{2}+\alpha_{\perp}^{2}}}+\frac{2\alpha_{\|}x_0}{(\alpha_{\|}^{2}+\alpha_{\perp}^{2})^{3/4}}+x_0^2}\right)\right. \\
&\left.+\rho\left(\sqrt{\frac{1}{\sqrt{\alpha_{\|}^{2}+\alpha_{\perp}^{2}}}-\frac{2\alpha_{\|}x_0}{(\alpha_{\|}^{2}+\alpha_{\perp}^{2})^{3/4}}+x_0^2}\right)\right\} \qquad (4.101)
\end{aligned}$$

$$\begin{aligned}
\Im\, C\left(u_{\|},\, u_{\perp}\right) =& -\frac{\pi}{2}\int \frac{d\alpha_{\|}\,\alpha_{\perp}d\alpha_{\perp}}{\left(\alpha_{\|}^{2}+\alpha_{\perp}^{2}\right)^{9/4}}\sin\left(u_{\|}\alpha_{\|}\right)J_0\left(u_{\perp}\alpha_{\perp}\right) \\
&\times\left\{\rho\left(\sqrt{\frac{1}{\sqrt{\alpha_{\|}^{2}+\alpha_{\perp}^{2}}}+\frac{2\alpha_{\|}x_0}{(\alpha_{\|}^{2}+\alpha_{\perp}^{2})^{3/4}}+x_0^2}\right)\right. \\
&\left.-\rho\left(\sqrt{\frac{1}{\sqrt{\alpha_{\|}^{2}+\alpha_{\perp}^{2}}}-\frac{2\alpha_{\|}x_0}{(\alpha_{\|}^{2}+\alpha_{\perp}^{2})^{3/4}}+x_0^2}\right)\right\}. \qquad (4.102)
\end{aligned}$$

It is evident that

$$\Re\, C\left(u_{\|},\, u_{\perp}\right) = \Re\, C\left(-u_{\|},\, u_{\perp}\right)$$
$$\Im\, C\left(u_{\|},\, u_{\perp}\right) = -\Im\, C\left(-u_{\|},\, u_{\perp}\right)$$

Hence

$$P\left(\tilde{a}\right) = 2\tilde{a}\,\frac{\left(GM\right)^3}{\pi}\int_0^\infty du_{\|}\int_0^\infty du_{\perp}u_{\perp}\frac{\sin\left(\tilde{a}\tilde{u}\right)}{\tilde{u}}\,e^{-\Re C\left(u_{\|},\,u_{\perp}\right)}\sin\left[\Im\, C\left(u_{\|},\, u_{\perp}\right)\right] \qquad (4.103)$$

with use of eqs. (4.100), (4.101) and (4.102).

A Gaussian model

We will now introduce a Gaussian model of a cluster, which is defined by

$$\rho(r) = \gamma e^{-\lambda r^2} \tag{4.104}$$

$$\gamma = n_{max} \tag{4.105}$$

$$n_{max} = (\frac{\lambda}{\pi})^3 N\,. \tag{4.106}$$

Then from eq.(4.101) and eq.(4.102) it follows that

$$\Re\; C(u_{\|}, u_{\perp}) = \pi\gamma \int \frac{d\alpha_{\|}\alpha_{\perp}d\alpha_{\perp}}{(\alpha_{\|}^2+\alpha_{\perp}^2)^{9/4}}\{1-\cos(u_{\|}\alpha_{\perp})J_0(u_{\perp}\alpha_{\perp})\}e^{-\lambda\left[\frac{1}{\sqrt{\alpha_{\|}^2+\alpha_{\perp}^2}}+x_0^2\right]} \tag{4.107}$$

$$\times \cosh\left[\lambda\frac{2\alpha_{\|}x_0}{(\alpha_{\|}^2+\alpha_{\perp}^2)^{3/4}}\right] \tag{4.108}$$

$$\Im\; C(u_{\|}, u_{\perp}) = -\pi\gamma \int \frac{d\alpha_{\|}\alpha_{\perp}d\alpha_{\perp}}{(\alpha_{\|}^2+\alpha_{\perp}^2)^{9/4}}\sin(u_{\|}\alpha_{\perp})J_0(u_{\perp}\alpha_{\perp})e^{-\lambda\left[\frac{1}{\sqrt{\alpha_{\|}^2+\alpha_{\perp}^2}}+x_0^2\right]} \tag{4.109}$$

$$\times \sinh\left[\lambda\frac{2\alpha_{\|}x_0}{(\alpha_{\|}^2+\alpha_{\perp}^2)^{3/4}}\right] \tag{4.110}$$

4.9.4 Spherical harmonic approach

Recall eq.(4.88)

$$C\left(\tilde{\boldsymbol{u}}\right) = \frac{1}{2}\int d^3\tilde{\alpha}\,\tilde{\alpha}^{-9/2}\left\{1-e^{\imath\tilde{\boldsymbol{u}}\cdot\tilde{\boldsymbol{\alpha}}}\right\}\rho\left[\tilde{\alpha}^{-3/2}\tilde{\boldsymbol{\alpha}}+\boldsymbol{x}_0\right]\,. \tag{4.111}$$

We now introduce spherical polar coordinates rather than cylindrical coordinates:

$$\boldsymbol{x}_0 = x_0\hat{\mathbf{1}}_z$$

$$\tilde{\boldsymbol{\alpha}} = (\tilde{\alpha},\, \vartheta_{\alpha},\, \varphi_{\alpha})$$

$$\tilde{\boldsymbol{u}} = (\tilde{u},\, \vartheta_u,\, \varphi_u)$$

where the angles are measured relative to $\hat{\mathbf{1}}_z$. Then (see appendix C.1)

$$e^{\iota\tilde{\boldsymbol{u}}\cdot\tilde{\boldsymbol{\alpha}}} = 4\pi\sum_{l=0}^{\infty}\iota^l j_l\left(\tilde{u}\tilde{\alpha}\right)\sum_{m=-l}^{l}Y^*_{lm}\left(\vartheta_\alpha,\varphi_\alpha\right)Y_{lm}\left(\vartheta_u,\varphi_u\right)\ . \tag{4.113}$$

If $\rho\left(\boldsymbol{x}\right)$ is spherically symmetric, then it depends only on $x = |\boldsymbol{x}|$, and $\rho\left[\tilde{\alpha}^{-3/2}\tilde{\boldsymbol{\alpha}}+\boldsymbol{x}_0\right]$ depends on $\sqrt{\tilde{\alpha}^{-1}+2\tilde{\alpha}^{-3/2}\tilde{\boldsymbol{\alpha}}\cdot\boldsymbol{x}_0+x_0^2}$ and hence is axially symmetric in $\tilde{\boldsymbol{\alpha}}$ about $\boldsymbol{x}_0$. If ρ is expanded in spherical harmonics the result will be of the form

$$\rho\left[\tilde{\alpha}^{-3/2}\tilde{\boldsymbol{\alpha}}+\boldsymbol{x}_0\right] = \sum_{l=0}^{\infty}\rho_l\left(\tilde{\alpha},\,x_0\right)Y_{l0}\left(\vartheta_\alpha,\varphi_\alpha\right)\ . \tag{4.114}$$

Substitution of eqs. (4.113) and (4.114) into eq. (4.111) leads to an expression of two terms, the first of which is

$$C\left(\tilde{\boldsymbol{u}}\right)_1 = -\sqrt{\pi}\int d\tilde{\alpha}\,\tilde{\alpha}^{-5/2}\rho_0\left(\tilde{\alpha},\,x_0\right) \tag{4.115}$$

and the second of which is

$$\begin{aligned}C\left(\tilde{\boldsymbol{u}}\right)_2 = -2\pi\int d\Omega_\alpha d\tilde{\alpha}\,\tilde{\alpha}^{-5/2}\Bigg\{&\sum_{l=0}^{\infty}\iota^l j_l\left(\tilde{u}\tilde{\alpha}\right)\sum_{m=-l}^{l}Y^*_{lm}\left(\vartheta_\alpha,\varphi_\alpha\right)Y_{lm}\left(\vartheta_u,\varphi_u\right)\\ &\sum_{l'=0}^{\infty}\rho_{l'}\left(\tilde{\alpha},\,x_0\right)Y_{l'0}\left(\vartheta_\alpha,\varphi_\alpha\right)\Bigg\}\end{aligned} \tag{4.116}$$

or

$$\begin{aligned}C\left(\tilde{\boldsymbol{u}}\right)_2 &= -2\pi\sum_{l=0}^{\infty}\iota^l Y_{l0}\left(\vartheta_u,\varphi_u\right)\int d\tilde{\alpha}\,\tilde{\alpha}^{-5/2}j_l\left(\tilde{u}\tilde{\alpha}\right)\rho_l\left(\tilde{\alpha},\,x_0\right)\\ &= -\sqrt{\pi}\sum_{l=0}^{\infty}\iota^l\sqrt{2l+1}\,\mathrm{P}_l\left(\cos\vartheta_u\right)\int d\tilde{\alpha}\,\tilde{\alpha}^{-5/2}j_l\left(\tilde{u}\tilde{\alpha}\right)\rho_l\left(\tilde{\alpha},\,x_0\right)\end{aligned} \tag{4.117}$$

using the fact that

$$Y_{l0}\left(\theta,\phi\right) = \sqrt{\frac{2l+1}{4\pi}}\,\mathrm{P}_l\left(\cos\theta\right)\ .$$

Putting eqs. (4.115) and (4.117) together we obtain

$$\begin{aligned}C\left(\tilde{\boldsymbol{u}}\right) = &-\sqrt{\pi}\int d\tilde{\alpha}\,\tilde{\alpha}^{-5/2}\left\{j_0\left(\tilde{u}\tilde{\alpha}\right)-1\right\}\rho_0\left(\tilde{\alpha},\,x_0\right)\\ &-\sqrt{\pi}\sum_{l=1}^{\infty}\iota^l\sqrt{2l+1}\,\mathrm{P}_l\left(\cos\vartheta_u\right)\int d\tilde{\alpha}\,\tilde{\alpha}^{-5/2}j_l\left(\tilde{u}\tilde{\alpha}\right)\rho_l\left(\tilde{\alpha},\,x_0\right)\end{aligned} \tag{4.118}$$

or

$$C(\tilde{\boldsymbol{u}}) = \sqrt{\pi}\sum_{l=1}^{\infty} \iota^{l}\sqrt{2l+1}\,\chi_l(\tilde{u},\, x_0)\,\mathrm{P}_l(\cos\vartheta_u) \tag{4.119a}$$

$$\chi_0(\tilde{u},\, x_0) = \int d\tilde{\alpha}\,\tilde{\alpha}^{-5/2}\left\{j_0(\tilde{u}\tilde{\alpha}) - 1\right\}\rho_0(\tilde{\alpha},\, x_0) \tag{4.119b}$$

$$\chi_l(\tilde{u},\, x_0) = \int d\tilde{\alpha}\,\tilde{\alpha}^{-5/2} j_l(\tilde{u}\tilde{\alpha})\,\rho_l(\tilde{\alpha},\, x_0)\,, \quad l > 0\,. \tag{4.119c}$$

Because

$$j_l(x) = \frac{\sqrt{\pi}}{2}\sum_{n=0}^{\infty}\frac{(-1)^n}{n!\Gamma\left(n+l+\frac{3}{2}\right)}\left(\frac{x}{2}\right)^{l+2n}$$

it follows that

$$j_0(x) - 1 = -\frac{x^2}{6} + \mathcal{O}\left(x^4\right)$$

so that $\rho_0(\tilde{u}\tilde{\alpha})$ must be bounded as $\tilde{\alpha} \longrightarrow 0$ for the first integral in eq. (4.118) to exist. For $l > 0$ we have

$$j_l(x) = \frac{\sqrt{\pi}}{2}\frac{1}{\Gamma\left(l+\frac{3}{2}\right)}\left(\frac{x}{2}\right)^{l} + \mathcal{O}\left(x^{l+2}\right)$$

so that $\rho_1(\tilde{u}\tilde{\alpha})$ must go to zero faster than $\tilde{\alpha}^{1/2}$, and $\rho_l(\tilde{u}\tilde{\alpha})$ for $l > 1$ must be bounded, as $\tilde{\alpha} \longrightarrow 0$, for the integrals in the second line of eq. (4.118) to exist. In point of fact for clusters with a finite tidal radius it appears that there will be a cutoff for small $\tilde{\alpha}$ below which ρ_l will be zero.

The Gaussian model in spherical polar coordinates

We now reintroduce a Gaussian model of a cluster, which is defined, as above, by

$$\rho(\boldsymbol{x}) = n_{max}\, e^{-\lambda x^2} \tag{4.120}$$

$$n_{max} = \left(\frac{\lambda}{\pi}\right)^3 N\,. \tag{4.121}$$

Then

$$\begin{aligned}\rho\left[\tilde{\alpha}^{-3/2}\tilde{\boldsymbol{\alpha}} + \boldsymbol{x}_0\right] &= n_{max}\, e^{-\lambda\left(\tilde{\alpha}^{-1} + 2\tilde{\alpha}^{-3/2}\tilde{\boldsymbol{\alpha}}\cdot\boldsymbol{x}_0 + x_0^2\right)} \\ &= n_{max}\, e^{-\lambda\left(\tilde{\alpha}^{-1} + x_0^2\right)} e^{-2\lambda\,\tilde{\alpha}^{-3/2}\tilde{\boldsymbol{\alpha}}\cdot\boldsymbol{x}_0}\end{aligned} \tag{4.122}$$

From the Rayleigh expansion (see appendix C.1) we have

$$\begin{aligned} e^{\iota \boldsymbol{k}\cdot\boldsymbol{x}} &= \sum_{l=0}^{\infty} \iota^l \left(2l+1\right) j_l\left(kr\right) \mathrm{P}_l\left(\cos\gamma\right) \\ e^{-\boldsymbol{k}\cdot\boldsymbol{x}} &= \sum_{l=0}^{\infty} \iota^l \left(2l+1\right) j_l\left(\iota kr\right) \mathrm{P}_l\left(\cos\gamma\right) \\ &= \sum_{l=0}^{\infty} \left(-1\right)^l \left(2l+1\right) i_l\left(kr\right) \mathrm{P}_l\left(\cos\gamma\right) \end{aligned}$$

as the modified spherical Bessel function of the first kind obeys

$$i_l\left(x\right) = \iota^{-l} j_l\left(\iota x\right) .$$

Hence

$$\begin{aligned} e^{-2\lambda\,\tilde{\alpha}^{-3/2}\tilde{\boldsymbol{\alpha}}\cdot\boldsymbol{x}_0} &= \sum_{l=0}^{\infty} \left(-1\right)^l \left(2l+1\right) i_l\left(2\lambda x_0/\sqrt{\tilde{\alpha}}\right) \mathrm{P}_l\left(\cos\vartheta_\alpha\right) \\ &= \sum_{l=0}^{\infty} \left(-1\right)^l \left(2l+1\right) i_l\left(2\lambda x_0/\sqrt{\tilde{\alpha}}\right) \sqrt{\frac{4\pi}{2l+1}}\, Y_{l0}\left(\vartheta_\alpha\right) \end{aligned}$$

whence

$$e^{-2\lambda\tilde{\alpha}^{-3/2}\tilde{\boldsymbol{\alpha}}\cdot\boldsymbol{x}_0} = \sqrt{4\pi} \sum_{l=0}^{\infty} \left(-1\right)^l \sqrt{2l+1}\, i_l\left(2\lambda x_0/\sqrt{\tilde{\alpha}}\right) Y_{l0}\left(\vartheta_\alpha, \varphi_\alpha\right) . \tag{4.123}$$

When this is substituted into eq. (4.122) the result is

$$\rho\left[\tilde{\alpha}^{-3/2}\tilde{\boldsymbol{\alpha}} + \boldsymbol{x}_0\right] = \sqrt{4\pi}\, n_{max}\, e^{-\lambda\left(\tilde{\alpha}^{-1}+x_0^2\right)} \sum_{l=0}^{\infty} \left(-1\right)^l \sqrt{2l+1}\, i_l\left(2\lambda x_0/\sqrt{\tilde{\alpha}}\right) Y_{l0}\left(\vartheta_\alpha, \varphi_\alpha\right) \tag{4.124}$$

which is of the form (4.114) with

$$\rho_l\left(\tilde{\alpha},\, x_0\right) = \left(-1\right)^l n_{max} \sqrt{4\pi\left(2l+1\right)} e^{-\lambda\left(\tilde{\alpha}^{-1}+x_0^2\right)}\, i_l\left(2\lambda x_0/\sqrt{\tilde{\alpha}}\right) . \tag{4.125}$$

The functions $\rho_l\left(\tilde{\alpha}, x_0\right)$ go strongly to zero as $\tilde{\alpha} \longrightarrow 0$, and also go strongly to zero as $x_0 \longrightarrow \infty$. For $l \geq 1$ they go to zero as x_0 goes to zero or as $\tilde{\alpha} \longrightarrow \infty$ but $\rho_0\left(\tilde{\alpha}, x_0\right)$ goes to a nonzero constant as $x_0 \longrightarrow 0$ or $\tilde{\alpha} \longrightarrow \infty$.

Substitution of eq. (4.125) into eq. (4.119c) yields

$$\chi_l\left(\tilde{u},\, x_0\right) = (-1)^l \sqrt{4\pi\,(2l+1)}\, n_{max}\; e^{-\lambda x_0^2} \int d\tilde{\alpha}\, \tilde{\alpha}^{-5/2} \left\{ j_l\left(\tilde{u}\tilde{\alpha}\right) - \delta_{l0} \right\}\, i_l\left(2\lambda x_0/\sqrt{\tilde{\alpha}}\right) e^{-\lambda/\tilde{\alpha}}\,. \tag{4.126}$$

Setting $\tilde{\alpha} = t^{-2}$ in the we arrive at

$$\chi_l\left(\tilde{u},\, x_0\right) = 2\,(-1)^l \sqrt{4\pi\,(2l+1)}\, n_{max}\; e^{-\lambda x_0^2} \int_0^{\infty} dt\, t^2 \left\{ j_l\left(\tilde{u}/t^2\right) - \delta_{l0} \right\}\, i_l\left(2\lambda x_0 t\right) e^{-\lambda t^2}\,. \tag{4.127}$$

Chapter 5

Concluding remarks

5.1 Slowly-changing variables

In this thesis we performed both a standard and an automated ARMA analysis of the period and angle of stars in a globular cluster. We found that these variables could not be fit to ARMA models of order five or less. We suggested that fitting higher order ARMA models, or models such as GARCHMA which incorporate heteroskedasticity, may be possible. Continuation along either of these veins would require extensive work.

While our time series analysis was not successful, there is still reason to believe that the study of globular clusters using slowly-changing variables is a worthwhile endeavour. As was mentioned in the introduction, the Delaunay variables are often used in the study of planetary systems, and there is no reason to believe a similar set of variables would not be as successful when applied to globular clusters. If a continuation of this work using higher order ARMA models or GARCHMA models were successful, this could lead to the derivation of a new set of kinetic equations describing globular clusters, which would be a very desirable outcome indeed.

5.2 Force distribution

We also showed in this thesis that the commonly held belief that the Holtsmark distribution accurately describes the distribution of forces in a star cluster is incorrect. We discussed the properties of the real force distribution, and we explained why this distribution differs from the Holtsmark distribution. We then presented the beginnings of the derivation of a more accurate force distribution for a globular cluster.

Much work will be required to derive the form of a fully generalized force distribution for globular clusters. This is a fundamental property of globular clusters, and thus there are many motivations, to go along with those we mentioned in chapter 4, which encourage us to achieve a better understanding of its behaviour.

Appendix A

Derivation of the collisionless Boltzmann equation

Here we will present a brief derivation of the collisionless Boltzmann equation. We will refer to the six dimensional phase space of position and velocity for a single star as the μ space. The state of each star is represented by a point in the μ space of that star, and the state of the entire system is given by N points in the μ space [27], where N is the number of stars in the system.

Before developing the collisionless Boltzmann equation, we define the coordinates in the μ space in terms of one vector $\mathbf{w}$ so that

$$(\mathbf{x}, \mathbf{v}) = \mathbf{w} = (w_1, ..., w_6). \tag{A.1}$$

We can express the flow of these six dimensional points in the μ space as

$$\dot{\mathbf{w}} = (\dot{\mathbf{x}}, \dot{\mathbf{v}}) = (\mathbf{v}, \mathbf{a}). \tag{A.2}$$

Due to the fact that we are assuming close stellar encounters are unimportant, we know that stars will pass through the μ space smoothly. Thus $\dot{\mathbf{w}}$ must satisfy a continuity equation. This is given by

$$\frac{\partial f}{\partial t} + \sum_{i=1}^{6} \frac{\partial (f \dot{w}_i)}{\partial w_i} = 0. \tag{A.3}$$

Physically, the first term on the left hand side of this equation represents the rate of increase of stars into the particular volume being considered, while the second term on the left hand side is the rate at which stars exit this volume [17]. Obviously to have continuity the sum of these two terms must be zero.

In order to simplify this equation we first expand it slightly:

$$\frac{\partial f}{\partial t} + \sum_{i=1}^{6} \dot{w}_i \frac{\partial f}{\partial w_i} + f \sum_{i=1}^{6} \frac{\partial \dot{w}_i}{\partial w_i} = 0. \tag{A.4}$$

Taking the third term of the left hand side, we have

$$f \sum_{i=1}^{6} \frac{\partial \dot{w}_i}{\partial w_i} = f \sum_{i=1}^{3} \left(\frac{\partial v_i}{\partial x_i} + \frac{\partial a_i}{\partial v_i} \right). \tag{A.5}$$

The $(\partial v_i / \partial x_i)$ term must be zero since the velocity and position coordinates are independent, so we get

$$f \sum_{i=1}^{3} \frac{\partial a_i}{\partial v_i} = 0. \tag{A.6}$$

This is zero because a is independent of v. Thus we can eliminate the third term from the left hand side of eq.(A.4), and we have

$$\frac{\partial f}{\partial t} + \sum_{i=1}^{6} \dot{w}_i \frac{\partial f}{\partial w_i} = 0 \tag{A.7}$$

or

$$\frac{\partial f}{\partial t} + \sum_{i} a_i \frac{\partial f}{\partial v_i} + \sum_{i} v_i \frac{\partial f}{\partial x_i} = 0. \tag{A.8}$$

and we have arrived at the collisionless Boltzmann equation. It is a special case of the well known Liouville's theorem [27].

Appendix B

Asymptotic analysis of the Holtsmark distribution

Recall eq.(4.44) for the Holtsmark distribution

$$H(\beta) = \frac{2}{\pi\beta} \int_0^\infty x \sin x e^{-(x/\beta)^{3/2}} dx. \tag{B.1}$$

If we make the substitution $x = \beta t$, $dx = \beta dt$ we get

$$H(\beta) = \frac{2}{\pi} \beta \int_0^\infty dt \, e^{-t^{3/2}} \, t \, \sin \beta t. \tag{B.2}$$

B.1 The Holtsmark distribution for small β

For small values of β we have $\sin \beta t \approx \beta t$ in eq.(B.2), and thus we have

$$H(\beta) = \frac{2}{\pi} \beta^2 \int_0^\infty dt \, t^2 \, e^{-t^{3/2}} + \mathcal{O}\left(\beta^4\right) . \tag{B.3}$$

Setting $s = t^{3/2}$, $dt = \frac{2}{3} s^{-1/3} \, ds$, we find that

$$\begin{aligned} H(\beta) &= \frac{2}{\pi} \cdot \frac{2}{3} \beta^2 \int_0^\infty ds \, s^{-1/3} s^{4/3} \, e^{-s} + \mathcal{O}\left(\beta^4\right) \\ &= \frac{4}{3\pi} \beta^2 + \mathcal{O}\left(\beta^4\right) \end{aligned} \tag{B.4}$$

where

$$\frac{4}{3\pi} = 0.424413 \tag{B.5}$$

B.2 The Holtsmark distribution for large β

To estimate the Holtsmark distribution for large β, set $x = s^2$ in eq. (B.2). Then

$$\begin{aligned} H(\beta) &= \frac{4}{\pi\beta}\int_0^\infty ds\, s^3\, e^{-s^3/\beta^{3/2}}\, \sin s^2 \\ &= \frac{2}{\iota\pi\beta}\int_0^\infty ds\, s^3 \,\exp\left(-s^3/\beta^{3/2} + \iota s^2\right) + c.c.\,. \end{aligned} \tag{B.6}$$

Consider

$$I_\pm = \int_0^\infty ds \,\exp\left(-s^3/\beta^{3/2} \pm \iota s^2\right)\ . \tag{B.7}$$

Set $s = \alpha t$ with $\alpha = \beta^{3/2}$. Then

$$I_\pm = \alpha\int_0^\infty dt\, exp\left[-\beta^3\left(t^3 \pm \iota t^2\right)\right]\ . \tag{B.8}$$

This is heading in the direction of a saddle–point expansion in the manner of Airy functions.

Let us take a different tack. Write eq. (B.2) in the form

$$H(\beta) = \frac{1}{\iota\pi\beta}\int_0^\infty dx\, x\, \exp\left[-\left(x/\beta\right)^{3/2} + \iota x\right] + c.c. \tag{B.9}$$

If this is regarded as a complex integral then the path of integration can be deformed from the real axis to the path

$$\begin{aligned} z &= r\, e^{\iota\delta} \\ 0 &\le r < \infty \\ 0 &< \delta < \pi/3 \end{aligned} \tag{B.10}$$

Along this path the factor $e^{\iota z}$ will provide convergence. Hence expand $\exp\left[-\left(z/\beta\right)^{3/2}\right]$ to obtain

$$\exp\left[-\left(z/\beta\right)^{3/2}\right] = 1 - z^{3/2}\beta^{-3/2} + \cdots . \tag{B.11}$$

Keep only the first terms in this expansion, and deform the path of integration to the positive imaginary axis: $z = re^{\iota\pi/2}$. Setting

$$\begin{aligned} H\left(\beta\right) &= \frac{1}{\iota\pi\beta} e^{\iota\pi/2} \int_0^\infty dr \left(re^{\iota\pi/2} - e^{3\iota\pi/4} t^{5/2}\beta^{-3/2}\right) e^{-r} + c.c. \\ &= \frac{-1}{\iota\pi\beta} e^{\iota\frac{5\pi}{4}} \Gamma\left(\frac{7}{2}\right) \beta^{-3/2} + c.c. \\ &= \frac{1}{\pi} \beta^{-5/2} e^{\iota\frac{7\pi}{4}} \Gamma\left(\frac{7}{2}\right) + c.c. \\ &= \frac{15}{2^{5/2}\pi} \Gamma\left(\frac{1}{2}\right) \beta^{-5/2} \end{aligned} \tag{B.12}$$

whence

$$H\left(\beta\right) = \frac{15}{8}\sqrt{\frac{2}{\pi}}\,\beta^{-5/2} + \mathcal{O}\left(\beta^{-4}\right) \quad \text{as} \quad \beta \to \infty . \tag{B.13}$$

Appendix C

Useful expansions, theorems and representations

C.1 Rayleigh's expansion

If $\boldsymbol{k} = (k,\, \vartheta',\, \varphi')$ and $\boldsymbol{x} = (r,\, \vartheta,\, \varphi)$ then Rayleigh's Expansion can be written as

$$\begin{aligned}
e^{\iota \boldsymbol{k}\cdot\boldsymbol{x}} &= 4\pi \sum_{l=0}^{\infty} \iota^l j_l\left(kr\right) \sum_{m=-l}^{l} Y_{lm}^{*}\left(\vartheta, \varphi\right) Y_{lm}\left(\vartheta', \varphi'\right) \\
&= \sum_{l=0}^{\infty} \iota^l \left(2l+1\right) j_l\left(kr\right) P_l\left(\cos\gamma\right) && \text{(C.1)} \\
\cos\gamma &= \frac{\boldsymbol{k}\cdot\boldsymbol{x}}{kr} && \text{(C.2)} \\
\mathrm{P}_l\left(\cos\gamma\right) &= \sqrt{\frac{4\pi}{2l+1}}\, Y_{l,0}\left(\gamma\right) && \text{(C.3)}
\end{aligned}$$

C.2 Addition theorem for spherical harmonics

We have

$$\begin{aligned} \mathrm{P}_l\left(\cos\gamma\right) &= \frac{4\pi}{2l+1}\sum_{m=-l}^{l} Y^*_{lm}\left(\vartheta,\varphi\right) Y_{lm}\left(\vartheta',\varphi'\right) && \text{(C.4)} \\ \cos\gamma &= \cos\vartheta\cos\vartheta' + \sin\vartheta\sin\vartheta'\cos\left(\varphi-\varphi'\right) && \text{(C.5)} \end{aligned}$$

C.3 Integral representation of Bessel functions of the first kind

We have

$$J_m\left(x\right) = \frac{1}{2\pi}\int_0^{2\pi} e^{\iota x\sin\varphi - \iota m\varphi} d\varphi\,. \qquad \text{(C.6)}$$

Bibliography

[1] Unpublished notes of J. C. Lewis (1998-2003).

[2] Webpage: *Treecode Guide* URL: `http://www.ifa.hawaii.edu/~barnes/treecode/treeguide.html`, Accessed on 20/06/2006.

[3] Webpage: *Satellite Tool Kit*, URL: `http://www.stk.com`, Accessed on 20/12/2003.

[4] J. Hawkin. Globular Clusters: N-Body Simulations and Interpretations. Honour's Thesis, Memorial University of Newfoundland, 2004.

[5] Mathematica Help File *Time Series*, Wolfram Research, 2003.

[6] Webpage: *Levinson recursion*, URL: `http://en.wikipedia.org/wiki/Levinson_recursion`, Accessed on 03/02/2006.

[7] Personal Correspondence with Darren Glosemeyer, Wolfram research.

[8] Webpage: *Normal distribution*, URL: `http://en.wikipedia.org/wiki/Normal_Distribution`, Accessed on 08/11/2005.

[9] Webpage: *Google Scholar*, URL: `http://scholar.google.com`. Accessed on 29/07/2006.

[10] Webpage: *Gaussian quadrature*, URL: `http://en.wikipedia.org/wiki /Gaussian_quadrature`, Accessed on 08/10/2005.

[11] Correspondence with Anton Antonov, Mathematica newsgroup, 26/06/2006.

[12] Correspondence with Carl Woll, Mathematica newsgroup, 19/03/2005.

[13] J. Aarseth. *Gravitational N-Body Simulations.* Cambridge University Press, 2003.

[14] O. D. Anderson. *Time Series Analysis and Forecasting.* Butterworth & Co., 1976.

[15] Ardi, Eliani, Inagaki, and Shogo. Distribution of forces in gravitationally clustered system. *Publ. of the Astronomical Society of Japan*, 49, 1997.

[16] J. Binney and M. Merrifield. *Galactic Astronomy.* Princeton University Press, 1987.

[17] J. Binney and S. Tremaine. *Galactic Dynamics.* Princeton University Press, 1987.

[18] P. Broersen. Robust algorithms for time series models. ProRISC/IEEE CSSP98, 1998.

[19] P. Broersen. Facts and fiction in spectral analysis. *IEEE Transactions on Instrumentation and Measurement*, 2000.

[20] P. Broersen. Automatic spectral analysis with time series models. *IEEE Transactions on Instrumentation and Measurement*, 2002.

[21] P. Broersen and S. de Waele. On the accuracy of time series models for turbulence. 8th Int. EALA Conference on Laser Anemometry, 1999.

[22] P. Broersen and S. de Waele. Automatic identification of time-series models from long autoregressive models. *IEEE Transactions on Instrumentation and Measurement*, 2005.

[23] S. Chandrasekhar. Stochastic problems in physics and astronomy. *Reviews of Modern Physics*, 15, 1943.

[24] D. Chang. *Geometric derivation of the delaunay variables and geometric phases.* Kluwer Academic Publishers, 2003.

[25] C. Chatfield. *The Analysis of Time Series.* CRC Press, 6th edition, 2003.

[26] C. Chiosi, E. Nasi, G. Bertelli, and A. Bressan. Magellanic cloud star clusters: The problems of age determination, metallicity - age relationship and agb star luminosity function. *Astronomy and Astrophysics*, 165(1-2), 1986.

[27] A. Choudhuri. *The Physics of Fluids and Plasmas.* Cambridge University Press, 1998.

[28] J. D. Cryer. *Time Series Analysis.* PWS Publishers, 1986.

[29] G. R. Fowles and G. L. Cassiday. *Analytical Mechanics.* Brooks Cole, 7th edition, 2004.

[30] M. Fujisaki, H. Ljungqvist. Estimation of voice source and vocal tract parameters based on arma analysis and a model for the glottal source waveform. IEEE International Conference on ICASSP, 1987.

[31] W. A. Fuller. *Introduction to Statistical Time Series.* John Wiley & Sons, Inc., 2nd edition, 1996.

[32] H. Goldstein. *Classical Mechanics.* Addison-Wesley, 1949.

[33] A. Harvey. *The Econometric Analysis of Time Series.* Oxford, 1981.

[34] J. Holtsmark. *Ann. d. Phys.*, 58, 1917.

[35] H. Kandrup. Stochastic gravitational fluctuations in a self-consistent mean field theory. *Physics Reports*, 63, 1980.

[36] Pietronero L. and Mohayaee R. A cellular automaton model of gravitational clustering. *Physica A*, 323, 2003.

[37] R. L. Liboff and N. Rostoker, editors. *Kinetic Equations.* Gordon and breach science publishers, 1971.

[38] J. Milton and J. Arnold. *Introduction to Probability and Statistics.* Irwin McGraw-Hill, 1995.

[39] S. Pfalzner and P. Gibbon. *Many-Body Tree Methods in Physics.* Cambridge University Press, 1996.

[40] L Pietronero, M Bottaccio, R Mohayaee, and M Montuori. The holtsmark distribution of forces and its role in gravitational clustering. *J. Phys.: Condens. Matter*, 14, 2002.

[41] M. Riley. A statistical model for generating pronunciation networks. Proc. Int'l Conf. on Acoustic, Speech and Signal Processing, 1991.

[42] J. P. Saul, J. K. Triedman, M. H. Perrott, and R. J. Cohen. Respiratory sinus arrhythmia: time domain characterization using autoregressive moving average analysis. *Am J Physiol Heart Circ Physiol*, 268, 1995.

[43] L. Spitzer. *Dynamical Evolution of Globular Clusters.* Princeton University Press, 1987.

[44] G. Wilson. The estimation of parameters in multivariate time series models. *Journal of the Royal Statistical Society*, 35, 1973.