Made-to-measure galaxy models - I Methodology

R. J. Long\textsuperscript{1*} and S. Mao\textsuperscript{1†}
\textsuperscript{1}Jodrell Bank Centre for Astrophysics, Alan Turing Building, The University of Manchester, Manchester M13 9PL

Accepted 2010 January 27. Received 2010 January 26; in original form 2009 February 27

ABSTRACT
We re-derive the made-to-measure method of Syer & Tremaine (1996) for modelling stellar systems and individual galaxies, and demonstrate how extensions to the made-to-measure method may be implemented and used. We illustrate the enhanced made-to-measure method by determining the mass-to-light ratio of a galaxy modelled as a Plummer sphere. From the standard galactic observables of surface brightness and line-of-sight velocity dispersion together with the $h_4$ Gauss-Hermite coefficient of the line-of-sight velocity distribution, we successfully recover the true mass-to-light ratio of our toy galaxy. Using kinematic data from Kleyna et al. (2002), we then estimate the mass-to-light ratio of the dwarf spheroidal galaxy Draco achieving a V-band value of $539 \pm 136 \, M_\sun / L_\sun$. We describe the main aspects of creating a made-to-measure galaxy model and show how the key modelling parameters may be determined.

Key words: galaxies: kinematics and dynamics – galaxies: individual: Draco – methods: N-body simulations – methods: numerical

1 INTRODUCTION
Within the field of galactic and stellar dynamics, it has become common practice to model kinematic observations of a galaxy in order to interpret the observations and to understand better the underlying dynamical structures within the galaxy. N-body modelling is one of the techniques employed. Syer & Tremaine (1996) categorised methods for creating N-body systems into 3 groups namely distribution function based, moment or Jeans equation based and orbit based, placing their made-to-measure (M2M) method in a new particle based group. The M2M method is, however, similar to the methods in the orbit based group notably the method of Schwarzschild (1979). Schwarzschild’s method since its inception has undergone both significant development (for example, Chanamé et al. 2008) and exploitation (for example, Cappellari et al. 2007). Chanamé et al. (2008) extended the method to handle discrete stellar kinematic data sets, and Cappellari et al. (2007) used Schwarzschild’s method to model photometric and kinematic observations of elliptical galaxies from the SAURON survey. By comparison, Syer and Tremaine’s M2M method remained largely unutilised until the bulge and disk model of the Milky Way in Bissantz et al. (2004). More recently, the M2M method was enhanced by the inclusion of a capability to model kinematic data (Jourdeuil & Emsellem 2007, de Lorenzi et al. 2007 and de Lorenzi et al. 2008), and was applied to model the elliptical galaxies NGC3377 and NGC4697. More recently still, Dehnen (2009) implemented an alternative weight adaption mechanism within the M2M method.

In this paper we remain true to the outline in Syer & Tremaine (1996) but reframe the M2M method (section 2) slightly to improve the theoretical basis for the weight evolution equation, and introduce 2 further constraints - the first on the sum of the particle weights and the second on the isotropy of the velocity dispersion. We show how the parameters necessary to tune the model for a given situation may be determined (section 3). We use the method to determine the mass-to-light ratio of a toy galaxy (section 4) and then to estimate the mass-to-light ratio of the dwarf spheroidal galaxy Draco (section 5). We aim to provide sufficient detail and advice to enable others to produce their own implementation of the M2M method. Our implementation is described in section 6.

2 THE M2M METHOD
2.1 Outline
In brief, the M2M method is concerned with modelling stellar systems and individual galaxies as a system of test particles orbiting in a gravitational potential. Weights are associated with the particles and are evolved over time (many orbital periods) such that, by using these weights, observational measurements of a real galaxy are reproduced. The method uses these observational measurements as constraints on the model. Whilst it is tempting to
think of the particles as representing stars, the particles are more accurately described as phase space density elements (Hernquist & Ostriker 1992) whose motion is integrated along the characteristic curves of the collisionless Boltzmann equation. The gravitational potential may be prespecified or determined self-consistently, and may contain a dark matter component. Schwarzschild (1973) uses a similar approach but does not modify the weights during the main modelling run obtaining them instead at the end via linear programming.

Reviewed in the next section, section 2.2, are the theoretical approaches taken in Syer & Tremaine (1996) and de Lorenzi et al. (2007). We then recast the method from a maximum likelihood starting point paying particular attention to the origin of the derivative term in the weight evolution equation. The particle weight convergence analysis in Syer & Tremaine (1996) is not revisited in this paper.

2.2 Theory

The galactic observables used with a M2M model are moments of the distribution function and have the general form

$$Y_j = \int K_j(r, v) f(r, v) \, d^3 r \, d^3 v,$$

(1)

where subscript $j$ indicates an instance of the observable, $K_j$ is the kernel for the observation $Y_j$ and $f(r, v)$ is the phase space distribution function. In this context, typical observables are surface brightness and surface brightness times the luminosity weighted line-of-sight velocity dispersion squared. For a model of $N$ particles, this integral form is translated into

$$y_j(t) = \sum_{i} w_i(t) K_j(r_i(t), v_i(t)),$$

(2)

where index $i$ runs from 1 to $N$. Note that $K_j$ embodies a selection function such that only the particles which have a direct effect on the observation $y_j$ are included in the sum. The goal of the M2M method is to evolve the particle weights, $w_i(t)$, such that the time averaged model observations $y_j$ match the actual observations $Y_j$.

The weights, representing (in our case) the luminosity of individual particles implemented as a fraction of the total luminosity of the galaxy being modelled, are evolved using

$$\frac{d}{dt} w_i(t) = -\epsilon w_i(t) \sum_j K_j(r_i(t), v_i(t)) \Delta_j(t),$$

(3)

where index $j$ runs from 1 to $J$, $\Delta_j(t) = \frac{y_j(t) - Y_j}{\sigma(Y_j)}$, $Z_j$ is arbitrary, and the kernels $K_j$ are implemented by binning the model’s particle data. $\epsilon$ is small, positive and constant. Calculations of the $\Delta_j$ from the model are exponentially smoothed to reduce the impact of particle counting effects using

$$\frac{d}{dt} \tilde{\Delta}_j(t) = \alpha (\Delta_j(t) - \tilde{\Delta}_j(t)),$$

(4)

where $\alpha$ is small, positive and constant. $\Delta_j(t)$ is then replaced with $\tilde{\Delta}_j(t)$ in equation 3 by de Lorenzi et al. (2007) take the equivalent approach of smoothing $\Delta_j$.

The weight evolution equation may be extended in 2 ways - firstly by the introduction of a profit function $S$ to constrain the overall weight evolution and smooth observable reproduction (regularisation), and secondly by the inclusion of observational errors (de Lorenzi et al. 2007). The equation then becomes

$$\frac{d}{dt} w_i(t) = \epsilon w_i(t) \frac{\partial F}{\partial w_i(t)},$$

(5)

where

$$F = \mu S - \frac{1}{2} \chi^2,$$

(6)

and is to be maximised. $\epsilon$, $\alpha$ and $\mu$ in equations 3, 4 and 6 are discussed in more detail in Syer & Tremaine (1996). Here, the equivalent discussion is delayed until section 4.1. Note that to arrive at equation 3 from equation 5 the kernels must not depend on the particle weights. $S$, the profit function, varying known as the relative entropy or Kullback-Leibler divergence from some initial value (prior) of the particle weights, is given by

$$S = -\sum_i w_i(t) \ln \left( \frac{w_i(t)}{m_i} \right),$$

(7)

where the $m_i$ are the priors. $\chi^2$ is calculated as

$$\chi^2 = \sum_j \tilde{\Delta}_j^2,$$

(8)

with $\tilde{\Delta}_j$ being the exponentially smoothed form of either a relative difference between the model and target observations

$$\Delta_j = \frac{y_j - Y_j}{\sigma(Y_j)},$$

(9)

or, if $\sigma(Y_j)$ represents the measurement error in $Y_j$, in a more usual $\chi^2$ form

$$\Delta_j = \frac{y_j - Y_j}{\sigma(Y_j)}.$$

(10)

If multiple classes ($K$) of constraining observables are used, it is computationally convenient to replace $F$ by

$$F = \mu S - \frac{1}{2} \sum_k \chi_k^2.$$

(11)

The number of observations within each class may be different and may be subject to a different binning schemes. Our derivation of the weight evolution equation is very similar to that above and is based on constructing a likelihood function giving the likelihood of the model reproducing the measured galactic observations and then maximising it (in log form) subject to a time derivative constraint on the relative entropy of the weights.

Redefining $F$ now as

$$F = \frac{1}{2} \chi^2 - \frac{1}{\epsilon} \frac{d}{dt} \sum_i w_i(t) \ln \left( \frac{w_i(t)}{m_i} \right),$$

(12)

and substituting for $S$ gives

$$F = \frac{1}{2} \chi^2 - \frac{1}{\epsilon} \frac{d}{dt} \sum_i w_i(t) \ln \left( \frac{w_i(t)}{m_i} \right) - \mu \sum_i w_i(t) \ln \left( \frac{w_i(t)}{m_i} \right).$$

(13)
Maximising $F$ with respect to the weights gives the Syer & Tremaine (1996) weight evolution equation

$$
\frac{d}{dt}w_i(t) = -\epsilon_i w_i(t) \sum_j K_j(r_i(t), v_i(t)) \Delta_j(t) + \mu \left( \ln \left( \frac{w_i(t)}{m_i} \right) + 1 \right).
$$

(14)

The M2M method is similar to a maximum entropy method (for example, Richstone & Tremaine 1988) but in this case the prior reflects the need for the particle weights to be constant over time.

Additional observational or modelling constraints may be included simply by modifying $F$. Whether the weights will converge or whether the observations will be reproduced requires either experimentation or a convergence analysis to be performed as per Syer & Tremaine (1996). In general, constraints which are expressed as squared ‘distance measures’ appear to perform satisfactorily. A simple extension to the method which meets these caveats is to amend the $\chi^2$ term to take the form

$$
\frac{1}{2} \chi^2_{LM} = \frac{1}{2} \sum_k \lambda_k \chi^2_k.
$$

(15)

where the $\lambda_k$ are small positive parameters and may be used to rescale $\chi^2_k$, or to express the relative priority of observable class $k$ within the M2M model. Note that, while not expressing $\chi^2$ as in equation [15] de Lorenzi et al. (2007) employ a scaling factor, not dissimilar in function to the $\lambda_k$, in their ‘force of change’ equation (their $c'$).

Similarly, given that a weight is a fraction of the total galaxy luminosity $L$, that is

$$
\sum_i Lw_i = L,
$$

then it is not unreasonable to require that the method does not alter the total luminosity and an appropriate constraint to include is the following (minimisation) term in $F$

$$
- \frac{1}{2} \lambda_{\text{sum}} \left( \sum_i w_i - 1 \right)^2.
$$

(17)

Re-normalisation of the weights was considered and rejected as it would destroy the smoothing history built up in the $\Delta_j$. Syer & Tremaine (1996) and de Lorenzi et al. (2007) contain no such similar constraint. Dehnen (2009) has an alternative scheme for modifying $F$ to achieve weight conservation.

Discrete observables (for example, measurements of the line-of-sight velocities of individual stars) are incorporated into the method as follows. The probability, $p_{D,j}(x_{\perp,j}, v_{\parallel,j})$, of the model reproducing a discrete line-of-sight velocity distribution is found by convolving the line-of-sight velocity distribution (losvd) with a Gaussian incorporating the observational errors $\sigma_j$.

$$
p_{D,j} = \frac{1}{\sqrt{2\pi\sigma_j}} \int \text{losvd}(x_{\perp,j}, v_{\parallel}) \exp \left( \frac{(v_{\parallel} - v_{\parallel,j})^2}{2\sigma_j^2} \right) dv_{\parallel},
$$

(18)

where the $\parallel$ and $\perp$ subscripts indicate parallel and perpendicular to the line of sight. Equation [13] gains an additional constraint term

$$
L_D = \lambda_D \sum_j \ln(p_{D,j}),
$$

(19)

where $\lambda_D$ is a small positive parameter. Note that this is not the only way of incorporating discrete observables. For example, the $p_{D,j}$ could have been included directly in the log likelihood function used to create $\chi^2$.

From expressing the line-of-sight velocity distribution in terms of the distribution function

$$
\text{losvd}(x_{\perp,j}, v_{\parallel}) = \frac{\int dx_{\perp} dv_{\parallel} f(x, v)}{\int dx_{\perp} dv_{\parallel} f(x, v)},
$$

(20)

$p_{D,j}$ is calculated from the model as

$$
p_{D,j} = \frac{1}{\sqrt{2\pi\sigma_j}} \left( \sum_i N \delta_{ij} w_i \exp \left( -\frac{(v_{\parallel} - v_{\parallel,j})^2}{2\sigma_j^2} \right) \right),
$$

(21)

where the selection function $\delta_{ij}$ takes the value 1 if particle $i$ contributes to observation $j$ and is 0 otherwise. The contribution to the weight evolution equation is found by differentiating the log likelihood function $L_D$ with respect to the particle weights and independently exponentially smoothing the resulting numerator and denominator. de Lorenzi et al. (2008) arrive at an equivalent expression. Proper motion data could be incorporated into the method in a similar fashion but this is not explored further here.

If we require a model with an isotropic velocity dispersion, this may be achieved by defining a model observable

$$
y_j = \frac{\sum_i N Lw_i (2v^2_{\perp,i} - v^2_{\parallel,i}) \delta_{ij}}{\sum_i N Lw_i \delta_{ij}},
$$

(22)

where $v_r$ and $v_t$ are the radial and tangential velocities, and including

$$
- \frac{1}{2} \lambda_{\text{iso}} \sum_j y^2_j
$$

(23)

as a minimisation term in $F$. These expressions come directly from using luminosity weighted velocity dispersions in calculating the $\beta$ anisotropy parameter

$$
\beta = 1 - \frac{v_t^2}{2v^2_r}
$$

(24)

and then setting $\beta = 0$. The denominator $\sum_i N Lw_i \delta_{ij}$ either should be exponentially smoothed as per equation [11] to reduce particle counting effects, or alternatively may be replaced by an observationally derived value by recognising that $L_j = \sum_i N Lw_i \delta_{ij}$ is the luminosity of radial shell $j$. There is a third option which is to create a composite constraint of luminosity times the original constraint. The $L_j$ approach is used for the isotropy constraint in the rest of this paper. These 3 options apply not just to the isotropy constraint but to all constraints where a sum of weights appears in the denominator. To be precise, the isotropy constraint is a constraint on $\beta$ and it will not enforce strict isotropy with the same dispersion in each of the 3 velocity components $v_r$, $v_{\parallel}$ and $v_{\perp}$. Extending the constraint to the case where

**Made-to-measure galaxy models**
$\beta = \beta(r)$ is straightforward and has been implemented in [Dehnen (2009)].

2.3 Model observables and kernels

In this section we list the model observables and kernels from which we use an appropriate subset in the M2M models described in this paper. We show the selection function separately from the actual kernel and also abbreviate $K_j(r_j(t), v_j(t))$ to $K_{ji}$. Within a given class of observables, one particle contributes to only one observable. No attempts have been made to smear particles to mimic the effects of an observational point-spread function, say, as in [Syer & Tremaine (1996) and de Lorenzi et al. (2008)]. The total luminosity of the galaxy being modelled is $L$.

(i) Luminosity density

$$ y_j = \sum_i^{N} \delta_{ij} \frac{Lw_i}{V_j} $$

$$ K_{ji} = \frac{L}{V_j} $$

where $V_j$ is the volume of the associated model bin.

(ii) Surface brightness

$$ y_j = \sum_i^{N} \delta_{ij} \frac{Lw_i}{A_j} $$

$$ K_{ji} = \frac{L}{A_j} $$

where $A_j$ is the area of the associated model bin.

(iii) Surface brightness times luminosity-weighted line-of-sight second velocity moment

$$ y_j = \sum_i^{N} \delta_{ij} Lw_i \frac{v_j^2}{A_j} $$

$$ K_{ji} = \frac{Lv_j^2}{A_j} $$

(iv) Luminosity-weighted line-of-sight second velocity moment

$$ y_j = \sum_i^{N} \delta_{ij} Lw_i \frac{v_j^2}{A_j I_j} $$

$$ K_{ji} = \frac{Lv_j^2}{A_j I_j} $$

where $I_j$ is the measured surface brightness. This form of the observable is an alternative to that in item (ix).

(v) Surface luminosity times line-of-sight velocity distribution Gauss-Hermite coefficient ($n$)

$$ y_j = \sqrt{2\gamma^{-1} L} \sum_i^{N} \delta_{ij} w_i H_n(v_{norm,i}) \exp(-v_{norm,i}^2/2) $$

$$ K_{ji} = \sqrt{2\gamma^{-1} L} H_n(v_{norm,i}) \exp(-v_{norm,i}^2/2) $$

where $v_{norm,i} = (v_{i} - v_{best})/\sigma_{best}$, $\gamma$, $v_{best}$ and $\sigma_{best}$ are the line strength, the mean line-of-sight velocity and dispersion from the best Gaussian fit to the observed line-of-sight velocity distribution data. To determine the Hermite polynomial values, we use the recurrence relationship

$$ H_p(x) = \sqrt{\frac{2}{p}} x H_{p-1}(x) - \sqrt{\frac{p-1}{p}} H_{p-2}(x), \quad p \geq 2 $$

with $H_0(x) = 0$ and $H_1(x) = \sqrt{2} x$.

Modelling the line-of-sight velocity distribution with a truncated Gauss-Hermite polynomial series is discussed in [Gerhard (1993) and van der Marel & Franx (1993)].

We have not at this time implemented any schemes for deprojecting surface brightness to give a luminosity density distribution as in [Jourdain & Emsellem (2007)] or de Lorenzi et al. (2008) where a multi-Gaussian expansion (Emsellem et al. 1994) is used to construct the distribution for their elliptical galaxies. Deprojection is not an integral part of M2M method. If we require a luminosity density, we assume that deprojection could be performed and just use a theoretical luminosity density, modified as in section 3.2.

3 IMPLEMENTATION

We describe in this section some of the key practical issues to be addressed in designing and implementing a software system to meet the theoretical design in section 2.

3.1 Process and data flows

We have decomposed the process and data flows into 3 main phases. The first phase, preparation, covers creating the particle initial conditions, and either creating from theoretical functions the observational constraints to be used, or manipulating observations of real galaxies into a form suitable for modelling. The second phase is the actual running of the M2M model, and the third phase is concerned with the analysis of the output from the modelling run. The analysis phase is split into 3 components namely particle weight convergence, reproduction of the observational constraints and particle kinematics.

The execution phase follows quite naturally from the equations in section 2 and is shown in Figure 1. Given the number of particles used in modelling runs, in our case $5 \times 10^8 - 2 \times 10^9$, we have parallelised our implementation so that multiple computer processors may be used to reduce the overall execution elapsed times. We adopt a simple strategy whereby 1 processor controls the modelling run and is responsible for the collation and smoothing of the model produced observations and calculating the weight evolution constraint terms. The other processors are responsible for orbiting their subset of the particles, calculating their contribution to the model observations and updating their particle weights. At the end of a run, the control processor collates all the particle data for subsequent analysis. We find that this parallelisation strategy, without change, works acceptably well across a range of hardware platforms from single dual processor machines, clusters of PCs and on high performance computing systems with fast inter-processor data connections. As shown in Table 1, doubling the number of processors approximately halves the computer elapsed time. Note that our implementation does not as yet handle self-consistent potentials created by the particles themselves.

The weight convergence algorithm described in section 3.2 we have implemented in computer memory with the slave processors being responsible for monitoring the convergence
Table 1. MPI scaling

| Processors | Elapsed time (m) |
|------------|------------------|
| 8          | 18.3             |
| 16         | 8.0              |
| 32         | 4.0              |
| 64         | 2.0              |

The elapsed time figures are from a $5 \times 10^5$ particle model with 3 observational constraints run for 100 dynamical time units in a high performance computing environment.

In modelling galaxies, a system of large spatial extent, possibly infinite if theoretical functions are used, is being modelled in a computer system. Inevitably, model boundary effects occur primarily as a result of model observable values falling off faster with radius than would be the case with theoretical functions, and the limited number of particles close to the boundary. We solve this issue in one of two ways, either by the simple expedient of oversizing the models and only analysing the central portions, or by truncating the distribution function and constructing a tailored observable function. The distribution function is truncated by energy, limiting the energy to the range between the potential energy at the origin and at the model boundary, and setting the distribution function to zero for energies outside of this range. As [Kashlinsky (1988)] points out, this is not the only way of truncating the distribution function. For example, truncation by radius, resulting in an increase in the number of (bound) circular orbits, may be appropriate but we have conducted no experiments using this approach.

Assuming a distribution function which is a function of energy only, we construct the luminosity density function via a particle realisation of the truncated distribution function over the radial extent of the M2M model. To ensure the correct radial distribution of particles, we determine the density of particles with a given energy, $\nu_E(r)$, by integrating the distribution function over velocity space to give

$$\nu_E(r) = 4\pi f(E) \sqrt{2(E - \phi(r))}.$$  \hspace{1cm} (37)

We obtain the radial position for a particle by sampling uniformly randomly from the 'fraction within radius' function, $N_E(<r)$ given by

$$N_E(<r) = \frac{16\pi^2}{g(E)} \int_0^r \sqrt{2(E - \phi(r))r^2} dr.$$ \hspace{1cm} (38)

where $g(E)$ is the density of states function.

Figure 2 compares a luminosity density function created in this way with the usual analytic density function for a Plummer sphere (radii are in units of the core radius). As can be seen, the two functions match in the inner part of the model but with the constructed function going to zero as required at the boundary of the model. For data creation purposes, the constructed luminosity density function is used in tabular form and integrated numerically to create other luminosity related functions, for example surface brightness or luminosity weighted velocity dispersion.

Luminosity density is an example of an observable which decreases with radius. Modelling observables which increase with radius, for example a rising velocity dispersion, the effects are more extreme. As can be seen from Figure 3, the inner part of the model where there is a good match between the theoretical function and the particle realisation is reduced to $\approx 25\%$ of the radial extent of the model.

3.3 Particle initial conditions

All particles are given the same initial weight and prior equal to $1/N$, where $N$ is the number of particles in the model. For the particles' initial spatial and velocity coordinates we use one of three schemes,

(i) spatial coordinates allocated such that the particle dis-
distribution matches the luminosity distribution, and the velocity coordinates uniformly randomly distributed,

(ii) spatial coordinates allocated such that the particle distribution matches the luminosity distribution, and the velocity coordinates sampled randomly from a Gaussian distribution created from the velocity dispersion function (from solving the Jeans equations).

(iii) by assuming that the distribution function is a function of relative energy only, spatial and velocity coordinates obtained from from distributing the particles uniformly randomly in energy.

In the third scheme, we sample uniformly, randomly from the integrated differential energy distribution (see Binney & Tremaine 2008) to obtain a particle’s energy, use the energy to determine the maximum radius it implies, allocate the particle’s spatial position within that radius using \(\nu_\nu(r)\) (equation 37), and finally use the energy difference between between the particle’s energy and its spatial position to allocate the velocity components.

The first two schemes are appropriate when observational data from a real galaxy is being used, and the third scheme, when theoretical models are being used. All the schemes clearly need at least a potential to be specified in order to be used. We have however deliberately separated the creation of the initial conditions from the main modelling software to increase flexibility - we may choose to run the model with a different potential for example.

### 3.4 Observational constraints

Clearly any observations of a real galaxy will need to go through a process of manipulation and conversion in order to get them into a form where they can be used with a M2M model. For theoretical galaxy models, for the purposes of developing the M2M method, we create constraint data by sampling from a Gaussian distribution with mean the function value, and standard deviation calculated from a pre-specified relative error. We constrain data values so produced by limiting them to be within \(n\)-sigma of the function values (\(n\) is typically 2 or 3).

### 3.5 Weight convergence & observable reproduction

In assessing whether or not a modelling run has been successful, we require, amongst other criteria, a high degree of weight convergence and consistent observable reproduction over a number of orbits. In practice, we choose some representative time period \(T_c\) which will cover, for example, both the long and short period particle orbits. For weight convergence, we consider a particle’s weight to have converged if its maximum relative deviation from its mean weight over time period \(T_c\) is less than some predetermined model-wide tolerance, that is if

\[
\max_i \left| \frac{w_i - w_{\text{mean},i}}{w_{\text{mean},i}} \right| \leq \text{tolerance.} \tag{39}
\]

We take as our unit of time the local dynamical time at the half mass radius of the model

\[
1 \text{ time unit} = \sqrt{\frac{3\pi}{16G\bar{\rho}}}, \tag{40}
\]

where \(\bar{\rho}\) is the mean density inside the half mass radius. For a typical model run, we set \(T_c\) to be \(\approx 10\%\) of the model duration and use a tolerance of 5 per cent or less.

Good model stability with respect to the particle weights is also required. Stability in this context means the degree to which a model’s outputs vary if the particles are subsequently orbited with weight evolution turned off. This is particularly important if the particles are to be used in some further modelling process. Stability is affected by the number of particles with converged weights and the total weight associated with particles with unconverted weights, and is illustrated in section 4.

We consider a M2M model to have reproduced the actual observations if the smoothed model observations match the actual observations to within the measurement errors on the actual observations, that is if

\[
|\tilde{y}_j - Y_j| \leq \sigma(Y_j). \tag{41}
\]
\( \hat{y}_j \) may be calculated directly from \( \hat{\Delta}_j \) using the smoothed form of equation (10). Note that observable non-reproduction does not necessarily imply that the method has failed - the smoothness of the constraining data needs to be taken into account as does the degree of smoothing employed in the model.

Given our success criteria for a model run and the mechanisms we use to measure whether they have been achieved, we do not deploy a ‘phase mixing’ phase as described in de Lorenzi et al. (2007).

3.6 Numerical methods

For orbit integration, we use the standard interleaved second order leapfrog method (drift, kick, drift) with either a fixed or adaptive time step. We either specify the time step directly or allow the model to determine it from the local dynamical time at the origin. For adaptive time stepping we use a 3 level model, progressively decreasing a particle’s time step by a factor of 2 as it nears the origin. Over the duration of a typical modelling run of 250 dynamical time units, we achieve an average maximum relative energy precision \( \Delta E(t)/E(0) \approx 10^{-4} \) which represents a reasonable level of energy conservation and is satisfactory for M2M purposes. We have tried (and discarded pending further investigation) the ‘dimensionless time’ orbit integration noted in Dehnen (2004). While it does increase the numbers of orbits of particles with long orbital periods, we find that weight convergence is reduced by \( \approx 2\% \) and that there is no change regarding the type of particles with unconverged weights at the end of a modelling run - it remains the highest energy particles.

For integration of the weight evolution equation and exponential smoothing we use Euler’s method. For example, the exponential smoothing equation (equation (4)) becomes

\[
\hat{\Delta}_j(t + 1) = \hat{\Delta}_j(t) + \alpha \Delta t \left[ \Delta_j(t) - \hat{\Delta}_j(t) \right].
\]

(42)

Rather than updating the particle weights every time step, we take the weight evolution time step as an integer multiple (\( \leq 5 \) for production runs) of the orbit integration time step. By reducing the inter-processor data traffic, a saving in computer elapsed time is achieved with no loss in effectiveness of the M2M method. For example, changing the multiple from 1 to 5 results in \( \approx 50\% \) saving in elapsed time with minimal changes in both weight convergence and the model \( \lambda_{LM}^2 \) value.

For three dimensional radially dependant observables, we use a radial binning scheme, and a polar scheme, with azimuthal binning if required, for two dimensional observables. The bin sizes may be either regular or irregular. Also, individual bins are only used if required by the distribution of the observational data - that is, we allow for gaps in the data. For the regular schemes, we use uniform or logarithmic bin sizes, or, to assist in the resolution of any central density peak, pseudo-logarithmic radial bin sizes as described in Sellwood (2003).

\[
r_i = (r_{\text{max}} + 1)^{1/B} - 1, \quad i = 1, \cdots, B
\]

(43)

where \( r_{\text{max}} \) is the maximum radius, divided into \( B \) bins, and \( r_i \) is the radius of the \( i^{th} \) bin boundary. The software for binning, and also that for numerical integration and interpolation, utilises the GNU Scientific Library.\(^1\)

4 CREATING A M2M MODEL

In this section, we consider various aspects of creating a M2M model and cover tuning its key parameters, the impact of different particle initial conditions, how many particles to use, and modelling incomplete data sets. Section (1.2) deals specifically with the impact of the relative entropy derivative constraint.

4.1 Parameter tuning

The parameters are the weight convergence rate \( \epsilon, \alpha \) for exponential smoothing, \( \mu \) governing regularisation, the observable constraint \( \lambda_{\text{conn}} \)’s from equation (15) and \( \lambda_{\text{sum}} \) and \( \lambda_{\text{iso}} \) for the sum of weights constraint and isotropic dispersion constraint. The parameters are treated as tunable with their values being determined prior to the start of a modelling run. Setting the parameters should be thought of as a process - we have identified no simple mechanism which will determine all the parameters required in one trial modelling run. The key to the process is to establish the initial values for the parameters. Having done this, it is straightforward to perform a series of modelling runs, increasing or decreasing the parameter values, to achieve a particular desired position.

We first explain how we assess the impact of the relative entropy (regularisation) term in the weight evolution equation. To recap, where constructed observable data are being used, we create the data using Gaussian sampling from the various functions of the associated mathematical model. The combination of relative error and sigma cut-off determine the spread of data points around the functions. In practice, what is needed from a M2M model, no matter whether the data is constructed or real, is that the observable data points are individually reproduced but that a smooth curve approximating the data points and reflecting any key features in the data is generated. Given the method of data construction, the smooth curve approximating the data points should be the function the data were generated from. To give a measure of how effective a M2M model is in re-creating the underlying functions, the following relative sum of squares is calculated for every class \( s \) of observable constraint.

\[
C_s = \sum_j \left( \frac{X_j - \hat{X}_j}{\sigma_j} \right)^2
\]

(44)

where \( X_j \) is the theoretical value for data point \( j \) and \( x_j \) is the equivalent smoothed value produced by the M2M model \( x_j = Y_j + \hat{\Delta}_j \sigma(Y_j) \).

(45)

Where observations of a real galaxy are being used, the underlying functions will not be known. The solution then is to create a trial data set approximating the real data set, with known underlying functions, and to use that trial data to determine the degree of regularisation required (the value of

\(^1\) http://www.gnu.org/software/gsl/
Clearly, the evolution factors will have a range of values per-
suitable compromise. In this paper, taking the value at 2 effective radii gives a
haps spanning several orders of magnitude. For the models
is, the parameter is used to neutralise the evolution factor.
the observable constraint parameters by making the product
T change across the runs. This can be resolved by multiplying
balance of the terms in the weight evolution equation will
is taken as the maximum velocity occurring in the model.
and to the relative strengths of the observable constraint
low, that the observable
been achieved, that the amount of unconverged weight is
exercise are that a high degree of weight convergence has
section 2.3 and ignoring the total luminosity of the galaxy
equation.
From experience, we have found suitable start values for
the observable constraint parameters by making the product
of the typical evolution factor and parameter to be
≈ 10^-2. Setting the values initially to be lower than for
the observable constraints means that observable constraints
have a stronger influence in the weight evolution equation.

For the exponential smoothing parameter \( \alpha \), we perform
a series of modelling runs with no regularisation and with \( \alpha \)
being varied from 10^-3 (no smoothing, \( \alpha = 6t^{-1} \) - see equa-
tion [12] to 5 \times 10^{-3}. We find that, as the amount of expo-
nential smoothing increases, weight convergence increases,
the unconverged weight decreases and the magnitude of the
\( \chi^2_{LM} \) gradient increases. The statistical fluctuations in \( \chi^2_{LM} \)
are greatly reduced once 10^{-2} < \alpha < 10^{-1} where the exact
value depends on the number of particles being used (5 \times 10^{4}
at the lower end to 10^{5} at the upper). It therefore turns out
that the value of \( \alpha = 5.24 \times 10^{-2} \) is in fact a reasonable default value to use over quite a wide
range in particle numbers and bin configurations.

To summarise,

(i) Parameter determination must be viewed as a process.
(ii) For the main observable constraints, setting the prod-
uct of the typical evolution factor and parameter to be of
O(1) gives a means of determining the initial values of the
parameters. For other \( \lambda \) constraints, setting the product to
\approx 10^{-2} gives a usable start position.
(iii) For the other parameters, \( \epsilon = 0.025, \alpha = 0.05 \) and
\( \mu = 0.001 \) are reasonable starting values.
(iv) The larger the observable errors and the greater the
spread of data points, the more likely it is that smoothing/
regularisation from the relative entropy term will be needed
and a higher value of \( \mu \) required. Quite how much regula-
risation is required (or desirable) is application specific. With
increased regularisation it is highly likely that observable re-
production will reduce. Based on the experiments we have
conducted, a higher value of \( \mu \), changing the influence of
terms in the weight evolution equation, requires a reduction
in the value of \( \epsilon \) to achieve acceptable results (for example,
high weight convergence).
(v) Regardless of the value of \( \epsilon \), the rate of particle weight
convergence should be monitored. Usable results may be ob-
tained from shorter modelling runs.

As a final comment, particularly when comparing pa-
rameter values between papers, it is the \( \epsilon \) parameter prod-
uct (for example, \( \epsilon \mu \) in equation [22] which is important.
Parameter values may be rescaled as required provided \( \epsilon \) is
rescaled correspondingly. Also, determination of the param-
eter values may appear onerous but it should be remembered
that it is only necessary to include the constraints and pa-
rameters that are required. The \( \mu \) adjustment process has not been used but the process or
its equivalent may be applicable to other parameters and
requires further investigation.

4.2 Relative entropy and the derivative constraint

We investigate the behaviour of the relative entropy regu-
larisation term \( (S) \) and the relative entropy derivative con-
straint \( (dS/dt) \) as the \( \mu \) parameter is increased from 10^{-3} to
10^{10}. We perform 2 sets of runs, the first having no further
constraints and the second using the total particle weight
constraint. 5 \times 10^{4} particles are used and \( \lambda_{\text{sum}} = 7 \times 10^{2} \) in
the second set of runs. The theoretical maximum in $S$ occurs when $w_i = m_i/E$ at which time, given that $m_i = 1/N$ where $N$ is the number of particles, $S_{\text{max}} = 1/E$ and $\sum w_i = 1/E$. For higher value of $\mu$, these values for $S_{\text{max}}$ and $\sum w_i$ can be seen in Table 3 in the ‘unconstrained weight’ columns. However, there is a conflict between the value of $S$ and that the method is achieved after $\approx 750$ time units for the lower value of $\mu$ and after $\approx 100$ units for the higher value.

### Table 4. Particle initial conditions modelling parameters

| Parameter                        | Model Value |
|----------------------------------|-------------|
| Overall model size               | 10 units    |
| Number of particles              | $2 \times 10^5$ |
| Model duration                   | 250 units   |
| Orbit integration time step      | 0.02 units  |
| Weight convergence monitoring    | 25 units    |
| Weight convergence tolerance     | 5%          |
| $\epsilon$                      | $2.5 \times 10^{-3}$ |
| $\alpha$                        | $5.0 \times 10^{-2}$ |
| $\mu$                            | 1.0         |
| Surface brightness, $\lambda_{SB}$ | $5.0 \times 10^{-4}$ |
| Second velocity moment, $\lambda_{V2}$ | $5.0 \times 10^{-3}$ |
| Gauss-Hermite $h_4$, $\lambda_{h4}$ | $2.0 \times 10^{-3}$ |
| Sum of weights, $\lambda_{\text{sum}}$ | $7.0 \times 10^2$ |
| Isotropic dispersion, $\lambda_{\text{iso}}$ | $1.0 \times 10^{-1}$ |

Spatial distances are given in units of the projected half light radius and times or durations in units of the half mass dynamical time. 32 bins are used for the surface brightness constraint and 24 for the velocity related constraints.
Table 3. Impact of increasing regularisation

| $\mu$    | $-F$     | $S$     | $\chi_{LM}^2$ | $\chi_{SB}^2$ | $\chi_{V2}^2$ | Particles Converged (%) | $\sum w_i$ | $C_{SB}$     | $C_{V2}$     |
|----------|----------|---------|----------------|---------------|---------------|------------------------|------------|--------------|--------------|
| 0.0      | $5.48 \times 10^{-2}$ | $-9.03 \times 10^{-3}$ | 1.10 $10^{-1}$ | 2.27 $10^{1}$ | 1.88 $10^{1}$ | 99.53                  | 0.62       | 1.00         | $3.01 \times 10^{-2}$ | $1.63 \times 10^{-1}$ |
| 1.0 $10^{-3}$ | $5.48 \times 10^{-2}$ | $-9.02 \times 10^{-3}$ | 1.10 $10^{-1}$ | 2.27 $10^{1}$ | 1.88 $10^{1}$ | 99.53                  | 0.62       | 1.00         | $3.01 \times 10^{-2}$ | $1.63 \times 10^{-1}$ |
| 1.0 $10^{-2}$ | $5.49 \times 10^{-2}$ | $-8.95 \times 10^{-3}$ | 1.10 $10^{-1}$ | 2.28 $10^{1}$ | 1.88 $10^{1}$ | 99.53                  | 0.63       | 1.00         | $3.00 \times 10^{-2}$ | $1.63 \times 10^{-1}$ |
| 1.0 $10^{-1}$ | $5.61 \times 10^{-2}$ | $-8.63 \times 10^{-3}$ | 1.10 $10^{-1}$ | 2.30 $10^{1}$ | 1.90 $10^{1}$ | 99.54                  | 0.62       | 1.00         | $2.93 \times 10^{-2}$ | $1.57 \times 10^{-1}$ |
| 1.0 $10^{0}$ | $6.27 \times 10^{-2}$ | $-3.28 \times 10^{-3}$ | 1.17 $10^{-1}$ | 2.51 $10^{1}$ | 2.04 $10^{1}$ | 99.44                  | 0.72       | 1.00         | $2.43 \times 10^{-2}$ | $1.13 \times 10^{-1}$ |
| 1.0 $10^{1}$ | $5.00 \times 10^{-3}$ | $1.34 \times 10^{-2}$ | 1.40 $10^{-1}$ | 3.48 $10^{1}$ | 2.44 $10^{1}$ | 98.23                  | 1.96       | 0.99         | $1.23 \times 10^{-2}$ | $2.63 \times 10^{-2}$ |
| 1.0 $10^{2}$ | $-6.07 \times 10^{0}$ | $1.51 \times 10^{-1}$ | 2.64 $10^{-1}$ | 2.32 $10^{2}$ | 2.93 $10^{1}$ | 96.64                  | 3.53       | 0.88         | $3.99 \times 10^{-1}$ | $3.02 \times 10^{-1}$ |
| 1.0 $10^{3}$ | $-2.58 \times 10^{2}$ | $3.41 \times 10^{-1}$ | 2.31 $10^{0}$  | 3.13 $10^{3}$ | $1.44 \times 10^{2}$ | 99.03                  | 1.01       | 0.52         | $7.29 \times 10^{0}$  | $5.47 \times 10^{0}$  |

Increasing regularisation ($\mu$) with the total weight constraint applied and other constraints as in Table 4. $C_{SB}$ and $C_{V2}$ are defined as per equation (44). At higher values of $\mu$, the weight evolution equation becomes ‘unbalanced’. $S$ increases as do the constraint $\chi^2$ values.

Table 5. Comparison of different particle initial conditions

| Run      | $-F$     | $\chi_{LM}^2$ | $\chi_{SB}^2$ | $\chi_{V2}^2$ | Particles Conv (%) | $\sum w_i$ | $C_{SB}$     | $C_{V2}$     |
|----------|----------|---------------|---------------|---------------|-------------------|------------|--------------|--------------|
| Energy   | $6.60 \times 10^{-2}$ | $1.23 \times 10^{-1}$ | 2.20 $10^{1}$ | 2.20 $10^{1}$ | 98.25             | 1.96       | $2.54 \times 10^{-2}$ | $7.92 \times 10^{-2}$ |
| Gaussian | $8.25 \times 10^{-2}$ | $1.26 \times 10^{-1}$ | 2.47 $10^{1}$ | 2.23 $10^{1}$ | 97.04             | 1.96       | $2.87 \times 10^{-2}$ | $1.11 \times 10^{-1}$ |
| Random   | $2.09 \times 10^{-1}$ | $1.47 \times 10^{-1}$ | 5.35 $10^{1}$ | 2.30 $10^{1}$ | 94.86             | 2.05       | $1.43 \times 10^{-1}$ | $1.64 \times 10^{-1}$ |

Random indicates that the spatial distribution matches the luminosity density with random velocity components while Gaussian has the same spatial distribution as Random but velocity components calculated by Gaussian sampling using the velocity dispersion. Energy indicates that the particle energies have been calculated utilising the distribution function.

4.4 Number of particles

We compare the effect of running a M2M model with different numbers of particles, from $10^4$ to $10^6$, and show the results in Table 6. We use energy-based particle initial conditions and the same parameter settings as in section 4.3. For the constraints used, the M2M model performs well for all numbers of particles. As might be expected, increasing the number of particles increases $F$, reduces the model $\chi_{LM}^2$, increases particle weight convergence and reduces the total weight associated with particles with unconverged weights.

Once an acceptable level of behaviour has been achieved from a M2M model, there is little to be gained by just increasing the number of particles without altering some other aspect of the model (for example, improving the spatial coverage of the observational constraints). Running the model with a low number of particles is attractive because of the reduced computer run time - the $10^6$ particle run in Table 6 took $\approx 2$ minutes on a 2.8 GHz ‘dual core’ workstation. However, the weight stability of the application (running the model with weight evolution turned off) must be considered. The stability of a $10^6$ particle run is worse (more variation in the model outputs over time) than that of a $10^5$ or a $5 \times 10^5$ particle run.

Changing the particle initial conditions to the Gaussian velocity scheme shows only a minor degradation in model behaviour - weight convergence, for example, reduces by less than 0.5%.

5 APPLICATION - MASS-TO-LIGHT DETERMINATION

5.1 Overview

We illustrate a practical application of the M2M method by using it to determine the mass-to-light ratio of a simple spherical Plummer model (Plummer 1911). Assuming that mass follows light and that the mass-to-light ratio is constant, we create a data set, comprising surface brightness, line-of-sight velocity dispersion and $h_4$ Gauss-Hermite coefficient values, with a known mass-to-light ratio (5 in this case). We run our M2M implementation with different mass-to-light ratios and expect that the data mass-to-light ratio will be indicated by a minimum in the values of $-F$ and the model $\chi^2$ values.

5.2 Model and data preparation

We use constructed functions for luminosity density $\nu(R, x)$ and velocity dispersion $\sigma(R, x)$ as described in section 4.2. In the following, we take the total luminosity as 1, the gravitational constant $G = 1$, $Y$ as the mass-to-light ratio, $R$ as the projected radius and $r$ as the spherical radius. Spatial distances are given in units of the projected half light radius (= 1 for our model). The key Plummer model expressions we require are

(i) Surface brightness

$$I(R) = \int dx || \nu(R, x) |. \tag{52}$$
Table 6. Impact of increasing the number of particles

| Number Particles | \(-F\)       | \(\chi^2_{LM}\) | \(\chi^2_{SB}\) | \(\chi^2_{V2}\) | Particles Conv (%) | Uconv Weight (%) | \(C_{SB}\)   | \(C_{V2}\)   |
|------------------|-------------|-----------------|-----------------|-----------------|-------------------|-----------------|-------------|-------------|
| \(1 \times 10^4\) | \(7.20 \times 10^{-2}\) | \(1.29 \times 10^{-1}\) | \(3.01 \times 10^1\) | \(2.15 \times 10^1\) | 95.93            | 4.50            | \(2.36 \times 10^{-2}\) | \(1.18 \times 10^{-1}\) |
| \(2 \times 10^4\) | \(6.59 \times 10^{-2}\) | \(1.21 \times 10^{-1}\) | \(2.59 \times 10^1\) | \(2.07 \times 10^1\) | 97.58            | 2.67            | \(2.85 \times 10^{-2}\) | \(9.59 \times 10^{-2}\) |
| \(5 \times 10^4\) | \(6.31 \times 10^{-2}\) | \(1.18 \times 10^{-1}\) | \(2.52 \times 10^1\) | \(2.03 \times 10^1\) | 98.76            | 1.41            | \(2.51 \times 10^{-2}\) | \(1.20 \times 10^{-1}\) |
| \(1 \times 10^5\) | \(6.26 \times 10^{-2}\) | \(1.17 \times 10^{-1}\) | \(2.55 \times 10^1\) | \(2.02 \times 10^1\) | 99.20            | 0.96            | \(2.22 \times 10^{-2}\) | \(1.15 \times 10^{-1}\) |
| \(2 \times 10^5\) | \(6.27 \times 10^{-2}\) | \(1.17 \times 10^{-1}\) | \(2.51 \times 10^1\) | \(2.04 \times 10^1\) | 99.44            | 0.72            | \(2.43 \times 10^{-2}\) | \(1.13 \times 10^{-1}\) |
| \(5 \times 10^5\) | \(6.24 \times 10^{-2}\) | \(1.17 \times 10^{-1}\) | \(2.52 \times 10^1\) | \(2.04 \times 10^1\) | 99.54            | 0.58            | \(2.39 \times 10^{-2}\) | \(1.10 \times 10^{-1}\) |
| \(1 \times 10^6\) | \(6.23 \times 10^{-2}\) | \(1.17 \times 10^{-1}\) | \(2.57 \times 10^1\) | \(2.03 \times 10^1\) | 99.59            | 0.53            | \(2.28 \times 10^{-2}\) | \(1.09 \times 10^{-1}\) |

Increasing the number of particles increases \(-F\), reduces the model \(\chi^2_{LM}\), increases particle weight convergence and reduces the total weight associated with particles with unconverged weights.

![Figure 5](image1.png)

Figure 5. End of run weight evolution comparison for the energy-based (top panel) and random velocity (bottom panel) schemes for creating particle initial conditions. The weights for the high energy particles in the random velocity scheme evolve furthest from their start values.

![Figure 6](image2.png)

Figure 6. Time evolution of \(\chi^2_{LM}\) for the energy-based (top panel) and Gaussian velocity (bottom panel) schemes for creating particle initial conditions.

\[
\phi(r) = \frac{\Gamma}{(r^2 + 1)^{1/2}}. \tag{55}
\]

The observational constraints are surface brightness, surface brightness times line-of-sight velocity dispersion squared and surface luminosity times the \(h_4\) Gauss-Hermite coefficient. Surface brightness is limited to a radial extent of 8 units and the velocity constraints to 5. We bin all observables radially, with pseudo-logarithmic bin sizes \(\text{[Sellwood]}\ 2003\), utilising 32 bins for surface brightness and 24 bins for
for the error terms, \( \sigma(Y_j) \), we use a relative error of 5 per cent for surface brightness and 10% for line-of-sight velocity dispersion. For \( h_4 \), we use an absolute error of 0.015 which is consistent with the published SAURON rms error (de Zeeuw et al. 2002). We create the surface brightness and line-of-sight velocity dispersion data values as described in section 3.3 using a 2-sigma cut-off. For \( h_4 \), we just take the theoretical values (times the bin luminosity). In addition to the observational constraints, we also impose the total weight and the isotropic velocity dispersion constraints.

For the purposes of this mass-to-light illustration, the particle initial conditions are energy based as described in section 3.3 and are created using the mass-to-light ratio of the model run, not the ratio with which the data was created. The particle initial weights and priors are set as \( 1/N \) where \( N \) is the number of particles.

### 5.3 Results

We execute the M2M model, using the parameters in Table 7 for 9 different mass-to-light ratios and plot the resulting model \(-F, -S\) and \(\chi^2_{LM}\) values against mass-to-light ratio. As can be seen from Figure 7 these model values have a minimum at a mass-to-light ratio of \( \approx 5 \). We fit smooth curves to \(-F, -S\) and \(\chi^2_{LM}\) using cubic spline interpolation and determine that the minimum values occur at mass-to-light ratios of 4.91, 4.71 and 4.97 respectively. By removing the \(\chi^2_{LM}\) factors (the \(\lambda_k\) in equation 15) and rescaling the \(\chi^2\) curve such that the \(\chi^2\) minimum value is equal to the number of degrees of freedom (78 in this case), we establish the 1σ error bounds and give the model determined value of the mass-to-light ratio of the data set as 4.88 ± 0.21. To complete the \(\chi^2\) analysis, we show the individual observable \(\chi^2\) plots in Figure 8.

The total weight constraint is met (Table 8) and so is the velocity dispersion isotropy constraint except at \( \Upsilon = 2.5 \). Weight convergence is high, peaking close to the true mass-to-light ratio, and, no less important, the weight associated with particles with unconverged weights is low at \(< 2\%\) of the total particle weight.

By examining the particle weight distribution in velocity space, we investigate how the M2M method behaves given a mass-to-light value which does not match the observational data set. For our Plummer model with the x-axis orientated to the line of sight, plotting the weight distribution contours perpendicular to the line of sight (in the \(v_x - v_y\) plane) we would expect to see a circular pattern as in Figure 9 and also in the other two planes. The \( \Upsilon = 5 v_x - v_y \) plot (Figure 10) is as expected. However, the \( \Upsilon = 2.5 \) plot

### Table 7. Mass-to-light modelling parameters

| Parameter                  | Model Value |
|----------------------------|-------------|
| Data mass-to-light ratio   | 5           |
| Overall model size         | 10 units    |
| Number of particles        | \( 5 \times 10^4 \) |
| Model duration             | 250 units   |
| Weight convergence monitoring | 25 units   |
| Weight convergence tolerance | 5%          |
| \( \epsilon \)             | 2.5 \times 10^{-3} |
| \( \alpha \)               | 5.0 \times 10^{-2} |
| \( \mu \)                  | 1.0         |
| \( \lambda_{SB} \)         | 5.0 \times 10^{-4} |
| \( \lambda_{VD} \)         | 2.5 \times 10^{-3}/\Upsilon |
| \( \lambda_{h4} \)         | 2.0 \times 10^{-3} |
| \( \lambda_{num} \)        | 10^4        |
| \( \lambda_{iso} \)        | 5.0 \times 10^{-1} |
| Line of sight axis         | x axis      |

Spatial distances are given in units of the projected half light radius and times or durations in units of the half mass dynamical time.

Figure 7. End of run \(-F, -S\) and \(\chi^2_{LM}\) plotted against mass-to-light ratio \(\Upsilon\). The minimum values are at \( \Upsilon = 4.91, 4.71 \) and \( \Upsilon = 4.97 \) respectively.
is stretched in the $v_x$ direction whilst the $\Upsilon = 10$ plot is compressed. Given the maximum velocity for $\Upsilon = 2.5$ is less than that of the data set ($\Upsilon = 5$), it is reasonable to expect that the M2M model selectively increases the weights of particles to try and reproduce the data set line-of-sight velocity dispersion. Similarly for $\Upsilon = 10$, with a greater maximum velocity, the weights of particles are selectively reduced. The end result is the distorted particle weight distributions.

In Figure 8 we display plots for different mass-to-light ratios showing the model luminosity per unit energy $dL/dE$ compared with the theoretical function and the velocity dispersion constraint compared with the observable data. At lower mass-to-light ratios, calculating $dL/dE$ from the end-of-run particle table shows that it is increased at low energies and reduced at high energies by comparison with its initial values. The converse is true for higher mass-to-light ratios with $dL/dE$ being reduced at low energies and increased at high.

Table 8. Weight convergence and dispersion isotropy

| $\Upsilon$ | Particles Converged (%) | Unconv Weight (%) | $\sum w_i$ | $\beta(0)$ | $\beta$ grad |
|------------|-------------------------|-------------------|------------|------------|-------------|
| 2.50       | 93.5                    | 6.5               | 1.0        | +0.001     | -0.026      |
| 3.33       | 98.3                    | 1.7               | 1.0        | -0.003     | -0.011      |
| 4.00       | 98.3                    | 1.6               | 1.0        | -0.003     | -0.006      |
| 4.50       | 99.1                    | 0.9               | 1.0        | -0.009     | -0.002      |
| 5.00       | 98.7                    | 1.3               | 1.0        | -0.016     | +0.003      |
| 5.55       | 98.3                    | 1.6               | 1.0        | -0.014     | +0.002      |
| 6.25       | 98.3                    | 1.6               | 1.0        | -0.014     | +0.002      |
| 7.50       | 98.3                    | 1.6               | 1.0        | -0.017     | +0.006      |
| 10.0       | 97.8                    | 2.0               | 1.0        | -0.020     | +0.006      |

Weight convergence peaks close to the true mass-to-light ratio. The total weight constraint is met for all runs and the velocity dispersion is isotropic except for $\Upsilon = 2.5$.

5.4 Different constraints and initial conditions

Clearly there are other combinations of constraining observables and particle initial conditions we could have used. We now examine the effect of replacing surface brightness (run 9 in Table 10) by luminosity density as a constraint (run 12), using an alternative velocity dispersion constraint (run 14), changing the particle initial conditions from energy based to Gaussian velocity (run 15) and random velocity (run 16), and finally increasing the number of particles from $5 \times 10^4$ to $2 \times 10^5$ (run 10). For the alternative velocity dispersion constraint, we remove the surface brightness multiplier so
Figure 11. Reproduction of $dL/dE$ and velocity dispersion. The rows, from the top, are for $\Upsilon = 2.5$ and $\Upsilon = 5$. The solid circles are the model produced values. In the left hand panels, the dotted line is the theoretical function. In the right hand panels, the solid line indicates the observable constraint data without error terms. The dispersion constraint is not met for $\Upsilon = 2.5$. The M2M model is unable to increase the particle weights sufficiently to reproduce the constraint. The $dL/dE$ plots show that it is the high relative energy particles which are most obviously affected with the lower energy particle weights being adjusted to compensate. The converse arguments apply for mass-to-light ratios higher than the true ratio.

Table 9. Different constraints and initial conditions

| Run | $\chi^2_{LM}$ | $\chi^2$ | $\chi^2_{VD}$ |
|-----|---------------|----------|---------------|
| 9   | 4.97          | 4.88 ± 0.21 | 4.90 ± 0.27   |
| 12  | 4.96          | 4.88 ± 0.18 | 4.91 ± 0.26   |
| 14  | 5.61          | 5.37 ± 0.28 | 5.38 ± 0.33   |
| 15  | 4.99          | 4.92 ± 0.22 | 4.79 ± 0.28   |
| 16  | 4.80          | 4.65 ± 0.27 | 4.52 ± 0.27   |
| 10  | 5.00          | 4.89 ± 0.22 | 4.90 ± 0.27   |

Run 9 is the main mass-to-light run described in section 5.5. Run 10 using 4 times as many particles as run 9 shows little difference in the results. The remaining runs are as per section 5.5.

that the constraint is just line-of-sight velocity dispersion squared.

We find that the largest variations from the true mass-to-light ratio come from using the alternative dispersion constraint and the random velocity initial conditions. For the other runs, the true mass-to-light is within 1-sigma of the model estimates. Note that increasing the number of particles makes very little difference.

5.5 Summary

To conclude this section, we have demonstrated the M2M method in action determining the mass-to-light ratio of a galaxy modelled by a theoretical Plummer sphere model. All the constraining observables are simple combinations of observational measurements regularly taken of actual galaxies, that is surface brightness, line-of-sight velocity dispersion and the $h_4$ Gauss-Hermite coefficient and their associated errors. For our constructed data generated with a mass-to-light ratio of 5, we are able to use the M2M method in a variety of ways to estimate that ratio. Using $\chi^2_{VD}$, our best estimate is $4.91 ± 0.26$ with a spread of results from $4.52 ± 0.27$ to $5.38 ± 0.33$.

6 DRACO

6.1 Introduction

Draco is a dwarf spheroidal satellite galaxy of the local group located some 75 kpc from the Sun and is interesting, both cosmologically and astrophysically, in that there is no currently accepted explanation for its stellar kinematics with-
Made-to-measure galaxy models

Figure 10. Particle weight distribution in velocity space parallel to the line of sight for Υ = 2.5 (top), Υ = 5 (target, centre) and Υ = 10 (bottom). As described in section 5.3, the contours show stretching and compression in the \( v_x \) direction as the M2M model tries to match the supplied velocity dispersion constraint for Υ values less than and greater than the target value. The reason for the double peak in the Υ = 2.5 plot is not known.

out invoking dark matter. Wilkinson et al. [2002] describe a mathematical model for modelling dwarf spheroidal galaxies and Kleyna et al. [2002] apply the model to Draco. In this section, we use the M2M method with the Draco data from Kleyna et al. [2002] plus the isotropic velocity distribution model from Wilkinson et al. [2002] to determine the mass-to-light ratio of Draco. The data set comprises 159 line-of-sight stellar velocity measurements with their errors. The spatial distribution of the measurements is shown in Figure 12. Binning the data to create a second velocity moment shows that the moment is rising with increasing radius. Other key data taken from Kleyna et al. [2002] are the central velocity dispersion of 8.5 km s\(^{-1}\), the effective radius of 9.71 arcminutes (≈ 214 pc), and the central V-band surface brightness of \( 2.2 \times 10^6 \) L\(_{\odot}\) kpc\(^{-2}\). Following the analysis in Wilkinson et al. [2002] and elsewhere, the surface brightness data is taken to follow a spherical Plummer model.

For an isotropic velocity dispersion with a rising line-of-sight velocity dispersion curve, the key equations based on Wilkinson et al. [2002] are

(i) Relative potential
\[
\psi(r) = \frac{\psi_0}{(1 + r^2)^{\alpha/2}}
\]
where \( \alpha < 0 \).

(ii) Distribution function
\[
f(E) \propto |E|^{5/\alpha - 3/2}.
\]

(iii) Circular velocity
\[
v_{\text{circ}}^2 = v_0^2 \frac{r^2}{(1 + r^2)^{1+\alpha/2}}
\]
where \( \psi_0 = v_0^2/\alpha \). Given that the matter distribution is spherical, \( v_{\text{circ}} \) may also be expressed as
\[
v_{\text{circ}}^2 = \frac{GM(<r)}{r}.
\]

(iv) Surface brightness
\[
I(R) = \frac{I_0}{(1 + R^2)^{\gamma/2}}
\]
where \( R \) is the projected radius, and \( I_0 \) is the measured central surface brightness.

(v) Line-of-sight second velocity moment
\[
\sigma^2(R) = \frac{\sigma_0^2}{(1 + R^2)^{\gamma/2}}
\]
where \( \sigma_0 \) is the measured central line-of-sight value.

Figure 12. Spatial distribution of Draco velocity measurements from Kleyna et al. [2002].
Before using Kleyna’s Draco data, we examine the ability of the M2M method to determine the Wilkinson potential power law index $\alpha$ which best fits the data, and then we use the circular velocity to determine the mass it implies (via equations 58 and 59) and thus the mass-to-light ratio. We vary $\alpha$ in the same manner that the mass-to-light $\Upsilon$ ratio was varied in section 6.2. The results are captured in Table 10.

### Table 10. M2M determination of Draco potential power law

| Measure | $\alpha$ | Error bounds |
|---------|----------|--------------|
| $-F$    | -1.18    |              |
| $\chi_{LM}^2$ | -1.24    | +0.12, -0.16 |
| $-S$    | -0.96    |              |
| $\chi^2$ | -0.95    | +0.04, -0.06 |
| $\chi_{SB}^2$ | -0.96    | +0.03, -0.04 |
| $\chi_{VD}^2$ | -0.90    | +0.36, -0.35 |

The value of the potential power law index $\alpha$ determined from different M2M model outputs.

\[
\sigma^2_\alpha = \frac{3\sqrt{\pi} v_0^2 \Gamma(2 + \alpha/2)}{4(\alpha + 5)\Gamma(5/2 + \alpha/2)}. \tag{62}
\]

Spatial distance are in units of the effective radius of Draco, and velocities, in effective radii per 10\(^7\) years.

The role of the M2M method is to determine the value of $\alpha$ for values of the Wilkinson $\alpha$ parameter in the range $-1.5$ to $-0.5$ with the same model parameters as in the proof of concept exercise in section 6.2. The results are captured in Table 10. Weight convergence is high (> 99%) in all runs except for $\alpha = -1.5$ where it is slightly lower (96%). There

### 6.2 Proof of approach

Before using Kleyna’s Draco data, we examine the ability of the M2M method to determine the Wilkinson $\alpha$ parameter. Surface brightness and line-of-sight second velocity moment data are created using the functions in section 6.1 with 1% relative errors for surface brightness and 10% for the velocity moment. For this trial data set, $\alpha = -0.5$. We run the M2M models with $2 \times 10^5$ particles and for 250 time units. The model parameters have values $\epsilon = 2.5 \times 10^{-3}$, $\mu = 1$, $\lambda_{\text{min}} = 10^3$ and $\lambda_{\text{max}} = 7 \times 10^{-2}$. $\lambda_{\text{SB}}$ has the fixed value $\lambda_{\text{SB}} = 10^{-5}$ while $\lambda_{\text{VD}}$ is varied to accommodate the different maximum velocities in the models as $\alpha$ is varied (from $\alpha = -0.8$ to $\alpha = -0.2$). We find that the modelling runs give clear minima (close to the true $\alpha$ value) in the $\chi^2$ values, with $\alpha = -0.51$ from $\chi_{\text{SB}}^2$ and $\alpha = -0.49$ from $\chi_{\text{VD}}^2$. That is, the approach works!

### 6.3 Modelling Draco

Binining the Draco velocity measurements (159 in total), using equal interval projected radius bins, to create a set of second velocity moment data points and error terms gives the plots in Figure 13. As can be seen, the measurements are centrally clustered and the velocity moment data points at higher radii suffer from a lack of contributing measurements. Comparing Figures 13 and 14, the trial data has more data points than the Draco data (24 versus 7) and the spread of points reflects the generating function. The Draco data does not visibly reflect any underlying curve.

After some experimentation, we run the M2M models for values of the Wilkinson $\alpha$ parameter in the range $-1.5$ to $-0.5$ with the same model parameters as in the proof of concept exercise in section 6.2. The results are captured in Figure 14. The trial second velocity moment data used in the proof of concept exercise in section 6.2.

![Figure 13. Draco second velocity moment from binned velocity measurements.](image1)

![Figure 14. The trial second velocity moment data used in the proof of concept exercise in section 6.2.](image2)
Table 11. Model vs theory line-of-sight velocity distribution

| \( \alpha \) | % within 5% | % within 10% |
|-------------|-------------|-------------|
| 1.50        | 66.0        | 88.7        |
| 1.25        | 84.3        | 100.0       |
| 1.10        | 89.3        | 100.0       |
| 1.00        | 93.7        | 100.0       |
| 0.90        | 99.4        | 100.0       |
| 0.75        | 88.1        | 98.1        |
| 0.50        | 81.8        | 90.6        |

For different values of \( \alpha \), the percentage of the individual velocity measurements that have model line-of-sight velocity distribution probabilities within 5% and 10% of their theoretical probabilities.

are no issues associated with the sum of weights constraint and the isotropic dispersion constraint.

We repeat the runs and include the individual velocity measurements as constraints (see section 2.2 equation 21) as well as the line-of-sight second moment. The individual measurements make very little difference to the results in Table 10. Taking the model produced line-of-sight velocity distribution and comparing it to the theoretical distribution at all of the individual velocity measurement points gives a measure of how well the M2M model has reproduced the theoretical distribution (Table 11). As can be seen the highest reproduction at the 5% level occurs at \( \alpha = -0.9 \) which is also the value resulting for \( \chi^2_{VD} \) in Table 10.

Comparing how well the model line-of-sight second velocity moment matches the theoretical moment for a given value of \( \alpha \) (Figure 15) shows that a good match is achieved for \( \alpha = -0.9 \). For \( \alpha = -0.5 \), the model overestimates the theoretical moment implying that the magnitude of \( \alpha \) needs to be increased, and for \( \alpha = -1.5 \) underestimates it implying that a reduction in the magnitude of \( \alpha \) is required.

Using \( \alpha = -0.96^{+0.36}_{-0.42} \), the mass within 3 core radii is \((9.7 \pm 2.3) \times 10^7 \text{ M}_\odot \). [Kleyna et al. 2002] achieved a value of \((8.1^{+1.2}_{-0.8}) \times 10^7 \text{ M}_\odot \). Taking Draco’s V-band luminosity as \(1.8^{+0.8}_{-1.0} \times 10^7 \text{ L}_\odot \) [Irwin & Hatzidimitriou 1995], the mass-to-light ratio for Draco is \(539 \pm 136 \text{ M}_\odot/\text{L}_\odot \). [Kleyna et al. 2002] obtained a lower value of \(440 \pm 240 \text{ M}_\odot/\text{L}_\odot \) which is not surprising given they were using an anisotropic dispersion model.

7 CONCLUSION

Other authors have commented on the potential of Syer \& Tremaine’s made-to-measure-method. We hope in this paper that we have helped to expose some of that potential in a practical way. We believe we have added clarity to the construction of the weight evolution equation, shown how constraints may be incorporated and defined 2 further constraints. We have demonstrated a simple application of M2M modelling by determining the mass-to-light ratio of a theoretical Plummer model. With a variety of constraints and initial conditions, we arrive in all cases at a model value close to the true value. Using the method with an isotropic velocity dispersion model, we estimate the mass-to-light ratio of Draco and achieve a V-band value of \(539 \pm 136 \text{ M}_\odot/\text{L}_\odot \).

We encourage others to use the method - particularly, for comparison purposes, those who have experience of using Schwarzschild’s method. It should be noted that much of the preparatory work (whether theoretical, or manipulation of observational data, say) leading to the execution of a Schwarzschild or a M2M model is in fact common. The key difference in the methods is how the final particle weights are determined. Also, the orbit selection stage in a Schwarzschild model is not required for a M2M model. The only shortfall against Schwarzschild’s method we are aware of is the modelling of proper motion data.
We have given insight into how a made-to-measure model may be implemented and provided some information on its behaviour and how to size and tune such a model. We have defined a simple mechanism for assessing the degree of particle weight convergence and are not aware of such a mechanism being used elsewhere with M2M models. From a software perspective, our first unparallelised implementation was less than 2000 lines of C code so the effort required to establish a working prototype is not huge.

As this paper was being completed, Dehnen (2009) became available and there is some overlap with this paper notably in total weight conservation and the use of the $\beta$ anisotropy parameter as a constraint.

Our next steps are to continue the move away from using theoretical models and apply the method to observations of more, real galaxies, and to understand practically the relative strengths of Schwarzschild’s method and the made-to-measure method.

ACKNOWLEDGEMENTS

RJL acknowledges receipt of an STFC postgraduate studentship. The authors would like to express their thanks to Scott Tremaine, James Binney and John Magorrian for various fruitful discussions, and to Wyn Evans for advice and guidance throughout.

REFERENCES

Binney J., Tremaine S., 2008, Galactic Dynamics: Second Edition, Princeton University Press
Bissantz N., Debattista V. P., Gerhard O., 2004, APJL, 601, L155
Cappellari M., Emsellem E., Bacon R., Bureau M., Davies R. L., de Zeeuw P. T., Falcón-Barroso J., Krajnović D., Kuntschner H., McDermid R. M., Peletier R. F., Sarzi M., van den Bosch R. C. E., van de Ven G., 2007, MNRAS, 379, 418
Chanamé J., Kleyna J., van der Marel R., 2008, ApJ, 682, 841
de Lorenzi F., Debattista V. P., Gerhard O., Sambhus N., 2007, MNRAS, 376, 71
de Lorenzi F., Gerhard O., Saglia R. P., Sambhus N., Debattista V. P., Pannella M., Méndez R. H., 2008, MNRAS, 385, 1729
de Zeeuw P. T., Bureau M., Emsellem E., Bacon R., Carollo C. M., Copin Y., Davies R. L., Kuntschner H., Miller B. W., Monnet G., Peletier R. F., Verolme E. K., 2002, MNRAS, 329, 513
Dehnen W., 2009, MNRAS, 395, 1079
Emsellem E., Monnet G., Bacon R., 1994, AAP, 285, 723
Gerhard O. E., 1993, MNRAS, 265, 213
Hernquist L., Ostriker J. P., 1988, APJ, 327, 82
Schwarzschild M., 1979, APJ, 232, 236
Sellwood J. A., 2003, APJ, 587, 638
Syer D., Tremaine S., 1996, MNRAS, 282, 223
van der Marel R. P., Franx M., 1993, APJ, 407, 525
Wilkinson M. I., Kleyna J., Evans N. W., Gilmore G., 2002, MNRAS, 330, 778
Kleyna J., Wilkinson M. I., Evans N. W., Gilmore G., Frayn C., 2002, MNRAS, 330, 792
Plummer H. C., 1911, MNRAS, 71, 460
Richstone D. O., Tremaine S., 1988, APJ, 327, 82

Kashlinsky A., 1988, APJ, 325, 566