A novel JEAnS analysis of the Fornax dwarf using evolutionary algorithms: mass follows light with signs of an off-centre merger

Foivos I. Diakogiannis, 1* Geraint F. Lewis, 2* Rodrigo A. Ibata, 3 Magda Guglielmo, 2 Prajwal R. Kafle, 1 Mark I. Wilkinson 4 and Chris Power 1

1International Center for Radio Astronomy Research, University of Western Australia, 35 Stirling Highway, Crawley, WA 6009, Australia
2Sydney Institute for Astronomy, School of Physics, A28, University of Sydney, NSW 2006, Australia
3Observatoire Astronomique, Université de Strasbourg, CNRS, 11, rue de l’Université, F-67000 Strasbourg, France
4Department of Physics & Astronomy, University of Leicester, Leicester LE1 7RH, UK

Accepted 2017 May 16. Received 2017 May 10; in original form 2016 July 27

ABSTRACT
Dwarf galaxies, among the most dark matter dominated structures of our Universe, are excellent test-beds for dark matter theories. Unfortunately, mass modelling of these systems suffers from the well-documented mass-velocity anisotropy degeneracy. For the case of spherically symmetric systems, we describe a method for non-parametric modelling of the radial and tangential velocity moments. The method is a numerical velocity anisotropy ‘inversion’, with parametric mass models, where the radial velocity dispersion profile, $\sigma_{rr}^2$, is modelled as a B-spline, and the optimization is a three-step process that consists of (i) an evolutionary modelling to determine the mass model form and the best B-spline basis to represent $\sigma_{rr}^2$; (ii) an optimization of the smoothing parameters and (iii) a Markov chain Monte Carlo analysis to determine the physical parameters. The mass-anisotropy degeneracy is reduced into mass model inference, irrespective of kinematics. We test our method using synthetic data. Our algorithm constructs the best kinematic profile and discriminates between competing dark matter models. We apply our method to the Fornax dwarf spheroidal galaxy. Using a King brightness profile and testing various dark matter mass models, our model inference favours a simple mass-follows-light system. We find that the anisotropy profile of Fornax is tangential ($\beta(r) < 0$) and we estimate a total mass of $M_{\text{tot}} = 1.613^{+0.050}_{-0.070} \times 10^8 M_\odot$, and a mass-to-light ratio of $\Upsilon_V = 8.93^{+0.32}_{-0.47} (M_\odot/L_\odot)$. The algorithm we present is a robust and computationally inexpensive method for non-parametric modelling of spherical clusters independent of the mass-anisotropy degeneracy.

Key words: methods: statistical – techniques: radial velocities – galaxies: dwarf – galaxies: individual: Fornax – galaxies: kinematics and dynamics – Local Group.

1 INTRODUCTION
Dwarf spheroidal galaxies (hereafter dSph) are some of the most dark matter (hereafter DM) dominated structures in our Universe. As such they are also some of the best laboratories for the search of DM. The nature of DM still eludes us: a discrepancy between theoretical predictions of the standard cold dark matter cosmological paradigm and observations is the so-called core-cusp problem. While numerical simulations predict cuspy profiles, predictions made from observations of dSph galaxies tend to favour cored profiles (Gilmore et al. 2007; Evans, An & Walker 2009). The debate is still open (Weinberg et al. 2015); therefore, we need methods that can potentially discriminate between these two categories. With respect to statistical modelling, the question of cuspy or cored haloes reduces to what is the best model fit (core or cusp) to given available data sets? Appropriate model inference methods need to be applied for the most reliable results.

1.1 Jeans modelling
A popular modelling technique for the estimation of DM profiles of dSph galaxies is the use of the spherically symmetric Jeans equation (Binney 1980). This method is known to suffer from the mass-velocity anisotropy degeneracy (MAD; Binney & Mamon 1982; see also Binney & Tremaine 2008; Merritt 2013). There has been extensive effort to break this degeneracy by many authors with
significant successes (e.g. Bicknell et al. 1989; Dejonghe & Merritt 1992; Lokas 2002; Mamon & Boué 2010; Wolf 2010, 2011; Mamon, Biviano & Boué 2013; Richardson & Fairbairn 2013; Ibata et al. 2013; Diakogiannis, Lewis & Ibata 2014a,b; Read & Steger 2017 and others; see chapter 5 Courteau et al. 2014 for a comprehensive review of MAD and mass modelling of galaxies in general).

In this contribution, we build on previous work (Diakogiannis et al. 2014a,b) and present the JEAnS algorithm (JEANS EVOLUTIONARY ANISOTROPY-FREE SOLVER) that performs mass modelling with the use of the Jeans equation independent of anisotropy, $\beta(r)$, assumptions. In fact, we nowhere use $\beta(r)$ in our modelling approach. The method is a velocity anisotropy ‘inversion’ in the sense that we calculate numerically the kinematic moments, $\sigma_{rr}^2$ and $\sigma_{\theta\theta}^2$, from assumptions on the mass models we use and the line-of-sight velocity dispersion, $\sigma_{\text{los}}^2$, data. However, we do not express $\sigma_{rr}^2$ and $\sigma_{\theta\theta}^2$ as functions of $\sigma_{\text{los}}^2$ as is done, e.g. in Binney & Merron (1982) or Dejonghe & Merritt (1992) (among others). A key ingredient of our method is to represent the unknown kinematic profile, $\sigma_{rr}^2$, with a B-spline curve\(^1\) (Diakogiannis et al. 2014a). Then we proceed in three distinct steps. In the first step, we use evolutionary optimization to select the best mass model and the B-spline basis for the representation of the radial velocity dispersion, $\sigma_{rr}^2$. In the second step, we optimize the smoothing parameters for the flexible B-spline representation of $\sigma_{rr}^2$ by using information from mock stellar systems. Finally, in the third step, we perform Markov chain Monte Carlo (MCMC) analysis in order to determine the physical parameters of the system under investigation with their uncertainties. Hence, we attack the problem of Jeans mass modelling in a twofold way: we model in a way that is independent of velocity anisotropy assumptions, and we discriminate between competing mass models. We will illustrate our algorithm by applying it to the well-sampled Fornax dSph.

1.2 Fornax dSph

Fornax is a classical dSph situated at a distance of approximately $d \approx 140$ kpc from Sun. The large number of measured heliocentric velocities that exist (Walker, Mateo & Olszewski 2009a) have allowed for robust statistical inference. However, it is still debatable which DM model (core or cuspy) best fits the observations. Various authors have presented mass estimates of Fornax, but this has a significant impact on the prediction of its mass content. Hence, we attack the problem of Jeans mass modelling in a twofold way: we model in a way that is independent of velocity anisotropy assumptions, and we discriminate between competing mass models. We will illustrate our algorithm by applying it to the well-sampled Fornax dSph.

In this contribution, we model Fornax independently of the MAD for a variety of DM mass models. We perform model selection using the modified Akaike criterion with second-order bias correction and present our findings.

1.3 Outline of the paper

In Section 2, we describe the mathematical formulation of the modelling approach used in the JEAnS algorithm. In Section 3, we describe in detail the three steps of the JEAnS. In Section 4, we describe the application of our method to a synthetic data set. In Section 5, we apply our method to the Fornax dSph and we report our results. Finally in Section 6, we present our conclusions and in Section 7, we describe possible future developments of our method.

Readers who are interested only in our results on the Fornax dSph can go directly to Section 5.

2 MATHEMATICAL FORMULATION

In this section, we describe in detail the mathematical formulation of the various objects we use in the JEAnS modelling approach. We start (Section 2.1) by giving a short description of B-spline functions that we use for the approximation of the radial velocity dispersion, $\sigma_{rr}^2$. Then we proceed (Section 2.2) in detailing the mass models as well as the necessary equations used in the JEAnS modelling approach. Finally (Section 2.3), we describe the likelihood function we use for the statistical analysis of the problem in the final step of the JEAnS.

2.1 B-spline functions

In previous work (Diakogiannis et al. 2014a), we have given a detailed description of B-spline functions. Here we summarize some key concepts for the convenience of the reader.

Loosely speaking, a B-spline function,\(^2\) $f(x)$, is a linear combination of some constant coefficients, $a_0$, with some polynomial functions $B_i(x)$ (B-spline basis functions) of a given degree $k - 1$, i.e. $f(x) = \sum a_i B_i(x)$. These basis functions are defined in a bounded interval, $[a, b]$, over a non-decreasing sequence of real numbers $\xi_0 \leq \xi_1 \leq \cdots \leq \xi_m = b$. The elements of this sequence, $\xi_i$, are called knots and their distribution and numbers regulate the geometric shape of the B-spline basis $B_i(x)$.

The B-spline basis forms a mathematical basis that is ‘as complete as possible’ for the approximation of real functions in a given interval. Then, for some suitable choice of coefficients, $a_0$, and B-spline basis, $B_i(x)$, a function $g(x)$ can be approximated by

$$g(x) \approx \sum_{i=1}^{n} a_i B_i(x). \quad (1)$$

The quality of this approximation depends crucially on the choice of the basis functions $\{B_i(x)\}$, i.e. on the order, $k$, of the basis and the choice of knot vector, $\{\xi_i\}$. In the following, the order of the B-spline polynomials will be fixed to $k = 5$, hence we define $B_i(x) := B_{5,i}(x)$. The best knot vector, $\{\xi_i\}$, will be a free parameter and will be evaluated with the use of evolutionary algorithms (hereafter EAs).

\(^1\) B-splines are flexible smooth curves that can take any geometric shape.

\(^2\) The C++ implementation of B-splines for this work can be found in https://github.com/feevos/bsplines.


2.2 Dynamical modelling

In this section, we describe our choice of mass models, the Jeans formalism we follow, as well as our B-spline approximation of the radial velocity dispersion, $\sigma_0^2$. In the following, we use the symbol $\theta_*$ for the set of stellar model parameters and $\theta_\star$ for the set of DM model parameters.

### 2.2.1 Mass models

In order to perform robust model selection and to include models from both cuspy and cored profiles, we use a variety of DM mass models, specifically Burkert (1995), to allow a cored DM profile suggested by many observations of spiral galaxy rotation curves, e.g. de Blok, McGaugh & Rubin 2001), NFW (Navarro et al. 1996, that represent well halo density profiles), Einasto (Einasto & Haud 1989, found by Navarro et al. 2004 to represent even better the halo profiles; see also Mamom & Lokas 2005 and Merritt et al. 2006) and a simple black hole (hereafter BH) potential. The functional form of the mass density of each of these is

$$
\rho_\star(r) = \begin{cases}
\rho_\star & \text{Burkert} \\
\frac{\rho_\star}{(1 + r/r_\star)(1 + (r/r_\star)^2)} & \text{NFW} \\
\frac{r_\star^3 \rho_\star}{r(r^2 + r_\star^2)} & \text{Einasto} \\
\frac{M_\star}{2\pi r_\star^2} \delta(r) & \text{Black hole}
\end{cases}
$$

where $\delta(r)$ is the Dirac delta function and $d_n = 3n - 1/3 + 0.0079/n$ (Merritt et al. 2006). For our stellar model, we use the mass density of the King (1966) profile that depends on the functional form of the potential, $\Phi(r)$. See Diakogiannis et al. (2014a) for the exact formalism we follow as well as Binney & Tremaine (2008) and references therein. The total potential of the system results from the solution of the Poisson equation:

$$
\nabla^2 \Phi = 4\pi G (\rho_\star + \rho_\star)
$$

In this way, the DM profile affects the solution of potential $\Phi$, and therefore the functional form of the stellar mass density, $\rho_\star$. Hence, the tracer projected mass density, $\Sigma(R)$, is influenced by the presence of DM and this is reflected in the brightness profile of the tracer population. This is a very important constraint on the available range of parameters for the stellar and DM models. This is also a very good approximation for modelling Fornax, since we expect that the stellar mass represents $\approx 10$ per cent of its total mass, i.e. the stellar component has a significant contribution to the total mass distribution.

### 2.2.2 The Jeans formalism

In the Jeans equation formalism, a stellar system is assumed to be fully described by the stellar (tracer) mass density, $\rho_\star$, the DM mass density, $\rho_\star$, the second-order radial, $\sigma_\star^2$, and tangential, $\sigma_\star^2$, velocity moments. The connection between these quantities and the line-of-sight velocity dispersion is given by the Jeans equation:

$$
-\frac{d \Phi}{dr} = \frac{d \rho_\star}{dr} + \frac{1}{\rho_\star} \frac{d \rho_\star}{dr} + \frac{2}{r} \sigma_\star^2 = \frac{\sigma_\star^2}{r}
$$

and the geometric definition of the projected line-of-sight velocity dispersion (Binney & Mamon 1982)

$$
\sigma_{los}^2 = \frac{1}{\Sigma(R)} \int_R^{\infty} \rho_\star \left[ \frac{2\sigma_\star^2(r^2 - R^2) + R^2 \sigma_\star^2}{r \sqrt{r^2 - R^2}} \right] dr,
$$

where $\Phi$ is the total potential resulting from both the stellar and DM mass distributions.

The connection between the velocity moments and the anisotropy profile is given through

$$
\beta(r) = 1 - \frac{\sigma_\star^2}{2\sigma_\star^2},
$$

where $\sigma_\star^2 = \sigma_\star^2 + \sigma_\star^2 = 2\sigma_\star^2 = 2\sigma_\star^2$.

### 2.2.3 Approximation of the radial velocity dispersion profile

At the heart of our method lies the approximation of the radial velocity dispersion, $\sigma_0^2$, with a B-spline function, i.e.

$$
\sigma_0^2(r) = \sum_{i=1}^{N_{\text{knots}}} a_i B_i(r).
$$

The choice of the particular basis, $B_i(r)$, as well as the constant coefficients $a_i$ regulates the geometric shape of the $\sigma_0^2(r)$ profile.

Once the basis functions, $B_i(r)$, are defined, the coefficients $a_i$ that regulate the geometric shape of $\sigma_0^2$ also regulate, in a linear way, the geometric shape of the line-of-sight velocity dispersion (Diakogiannis et al. 2014a,b)

$$
\sigma_{los}^2 = \sum_i a_i I_i(R) + C(R)
$$

where

$$
I_i(R) = \frac{1}{\Sigma(R)} \int_R^{\infty} 2r \rho_\star B_i(r) + R^2 \frac{d \rho_\star B_i(r)}{dr} \frac{dr}{\sqrt{r^2 - R^2}}
$$

and

$$
C(R) = \frac{1}{\Sigma(R)} \int_R^{\infty} \rho_\star R^2 \frac{d \Phi(r)}{dr} \frac{dr}{\sqrt{r^2 - R^2}}.
$$

Equation (8) results from solving equation (4) with respect to $\sigma_0^2$, and substituting under the integral sign of equation (5). The functions $I_i(R)$ and $C(R)$ depend on the stellar, $\rho_\star$, and DM, $\rho_\star$, mass densities. In addition, $I_i(R)$ depends also on the particular B-spline basis $B_i(r)$, which is a known function. Since equation (8) is linear with respect to the unknowns $a_i$, for a given set of parameters for the stellar and DM models, there exists a unique solution, $\hat{a}_i$, for the kinematic profile. The quality of the fit, however, depends on the particular choice of the knots, $\xi$, distribution and the smoothing penalty coefficient, $q$. Therefore, each set of parameters $\{\theta_\star, \theta_\star, \xi, q\}$ defines a unique model.

Using a B-spline approximation for the radial velocity dispersion, $\sigma_0^2$, we avoid any bias from a specific assumption on the functional form of the anisotropy profile. Effectively, we transform the set of all possible choices for the functional form of the anisotropy profile, $\beta(r)$, from a countable infinite discrete set to an infinite yet continuous set of $\sigma_0^2$ solutions. This continuity in the variation of the possible kinematic profiles significantly limits the number of models that we need to compare in order to choose the best one that closest approximates reality. The total number of models now is restricted to the total number of stellar tracer and DM model combinations.

3 Strictly speaking, a BH is not a DM profile; however, the mathematical formalism does not change.

4 See Section 3.3 for the definition of the smoothing penalty.
2.2.4 The need for adaptive knot distribution in JEAnS modelling

The B-spline approximation of a function, equation (1), depends on the quality of the data we have and the total number and geometric shape of the basis functions, $B_i(x)$. The latter are defined from the choice of the knot vector, $\xi$, and their order, $k$. A large knot vector results in a large number of basis functions and therefore a large number of unknown coefficients $a$. This increases the dimensionality of the unknown parameters and may result in overfitting. On the other hand, a bad choice of knots, $\xi$, can result in a bad approximation of the function – even if this is only locally. The problem of finding the optimum choice of knots, $\xi$, is a non-linear one and is generally very difficult to tackle. This is why in the literature the proposed method is to use a large number of uniform knots; then one regulates over-fitting or under-fitting with the use of a single smoothing parameter that is related to the penalty of the curvature of the fitted curve. See Hastie et al. (2001) and references therein for a description of the problem.

In order to get the best function approximation for $\sigma_n^2$ and $\sigma^{2 \alpha}$, with the least number of unknown coefficients, we need B-splines that are locally adaptive. This cannot be solely through the optimum smoothing parameter because it is a global regulator and cannot account for local irregularities in the data. This is more apparent when our data are incomplete with large errors (as is usually the case for a description of the problem.

One important property of B-spline functions, in the context of Galactic Dynamics, is their local support. In practice, this means that we can make local changes in the geometric shape of a curve without affecting its geometric shape globally. For example, when we fit a smoothing B-spline curve to data, the value of the curve at a given location, $x$, depends on the data in the local region around $x$. How ‘local’ the region around $x$ is depends crucially on the knot vector, $\xi$, that defines the basis $B_i(x)$. In the JEAnS formalism, the $\sigma^{2 \alpha}(R)$ profile is determined by two factors: the data and the smoothing penalty. The local support of the B-spline functions is inherited in the $\sigma^{2 \alpha}$ profile as well. Since our data extend up to a limited distance from the origin of the system, we want to have adaptive splines such that the unknown coefficients $a_i$ are all influenced by the data. In the extreme case where we have a very large number of unknowns, the value of the coefficients, $a_i$, that correspond to locations beyond the last datum will be solely determined by the smoothing penalty and result in a straight line.

Furthermore, when we want to compare two DM models, we would like to have kinematic profiles that are described in the least complex way. All the above make the evaluation of the optimum knot distribution an important part of our work. See Appendix A for a simple example on the significance and impact of the EA-best knot distribution chosen by our algorithm versus a uniform $\xi$ for simple curve fitting.

5 Wiki site http://astrowiki.ph.surrey.ac.uk/dokuwiki/. The performance of the JEAnS is currently tested with the GAIA CHALLENGE mock suite in Diakogiannis et al. (in preparation).

6 For a B-spline function of order $k$, $f(x) = \sum a_i B_i(x)$, at each location $x \in \{\xi_i, \xi_{i+1}\}$ there are at most $k$ non-zero basis functions, namely the $\{B_{i-k+1}(x), \ldots, B_{i}(x)\}$. This means that the corresponding coefficients that determine the geometric shape of the function at $x$ will be $\{a_{i-k+1}, \ldots, a_i\}$.

7 In fact, the virial radius of a system extends much further than the location of the last datum.

2.3 Statistical analysis

In this section, we describe our likelihood model, as well as the process for obtaining marginalized distributions of our model parameters. See Diakogiannis et al. (2014b) for an analytical justification of our choices of the various functional forms.

The general form of the posterior probability we use is

$$L = p(\theta | I) p(\tilde{q} | \alpha, \beta) p(W | \tilde{q}) L(D(\theta, I)).$$

(11)

For a fixed knot distribution, $\xi$, $p(\theta | I)$ is the prior range of all model parameters $\theta = \{\theta_1, \theta_2, \ldots, \alpha, \beta\}$, which we take to be uninformative (uniform). $p(\tilde{q} | \alpha, \beta)$ is the hyperprior on the smoothing parameter, $\tilde{q}$. $p(W | \tilde{q})$ is the smoothing penalty term and $L(D(\theta, I))$ is the likelihood of the model. In full detail, these equations are

$$p(\tilde{q} | \alpha, \beta) = \frac{\beta^a}{\Gamma(a)} \tilde{q}^{-a-1} e^{-\beta/\tilde{q}}$$

(12)

$$W = \int d^2 R \left( \frac{d \sigma^{2 \alpha}}{d^2 R} \right)^2$$

(13)

$$p(W | \tilde{q}) = \tilde{q} \exp(-\tilde{q} W)$$

(14)

$$\tilde{q} = \frac{1 - q}{q}, \quad q \in (0, 1]$$

(15)

and

$$L(D(\theta, I)) = \prod_{i=1}^{N_{\text{data}}} \exp \left[ -\frac{(\Sigma_i - \Sigma_*(R_i))^2}{2(\delta \Sigma_i)^2} \right] / \sqrt{2\pi (\delta \Sigma_i)^2}$$

$$\times \exp \left[ -\frac{(\sigma_i^2 - \sigma_{2 \alpha}(R_i))^2}{2(\delta \sigma_i)^2} \right] / \sqrt{2\pi (\delta \sigma_i)^2}.$$  

(16)

In equation (16), the variable $\Sigma_i$ is the surface density at position $R_i$, and $\delta \Sigma_i$ its uncertainty. Similarly, $\sigma_i^2$ is the observed line-of-sight velocity dispersion, and $\delta \sigma_i$ the uncertainty. We need to stress that the line-of-sight velocity distribution function (LOSVDF) is not Gaussian (Merritt 1987). In addition, even if the LOSVDF were Gaussian, the distribution function for the $\sigma_{2 \alpha}$ would follow a $\chi^2$ distribution that has much wider tails. Our likelihood in equation (16) should be viewed as the Bayesian approach of the weighted $\chi^2$ penalty function under the assumption of uncorrelated weights:

$$\chi^2 \sim \sum_i \left( \frac{\sigma_i^2 - \sigma_{2 \alpha}(R_i)}{\delta \sigma_i} \right)^2.$$  

The quantity $W$ is the smoothing penalty factor, while the function $\tilde{q}(q)$ is a convenient way of restricting the smoothing penalty parameter range to $(0, 1] \ni q$ for computational purposes. The prior on the $\tilde{q}$ values, $p(\tilde{q} | \alpha, \beta)$, is the inverse Gamma distribution, and the values of parameters $\alpha, \beta$ will be chosen from mock data sets of stellar systems in a process described later. This is a crucial part of our algorithm: we use information from theoretical models to identify the optimum smoothing value range for $\tilde{q}$. This is an extension of Empirical Bayes methods (Casella 1985) that defines prior information from moments of the data: here we encode some specific characteristic of ideal theoretical models (specifically the optimum smoothing of $\sigma_{2 \alpha}$) in order to build prior information.

8 In this contribution, we are using only the second-order derivative of $\sigma_{2 \alpha}$. In Diakogiannis et al. (2014b), we have used both first- and second-order derivatives. The main reason for our choice is computational efficiency.

9 See Diakogiannis et al. (2014b) for the justification of this choice of hyperprior.
3 THE JEANS ALGORITHM

In this section we describe the steps of the JEAnS algorithm. The method we propose is a three-phase method. The three phases are the following:

(i) In phase 1 (Section 3.1), we evaluate the optimum knot distribution and discriminate between competing mass models.

(ii) In phase 2 (Section 3.2), we encode information for the optimum smoothing of the B-splines from ideal theoretical models.

(iii) In phase 3 (Section 3.3), we use MCMC to estimate the marginalized distributions of the model parameters, keeping the knots fixed.

3.1 Evolutionary model selection

In this section, we will state the nature of the modelling problem and how our algorithm tackles it. Then we will describe the framework and the particular characteristics of the evolutionary algorithm we developed. For the convenience of the readers, we try to avoid as much as possible the standard terminology from the EA literature. In Appendix C, we give a small glossary of terms. Readers who wish to understand more about evolutionary algorithms can consult Goldberg (1989) for a first understanding and Talbi (2009) and Rozenberg, Bck & Kok (2011) for the state of the art in evolutionary computation.

3.1.1 The need for evolutionary algorithms

By approximating the radial velocity dispersion, $\sigma^2_{rr}$, with a B-spline function, we introduce a variable number of unknown variables: the knot vector $\xi$ as well as the unknown coefficients $a_i$. So the problem of fitting our model to a particular data set now becomes: find the best set of parameters for the stellar model, $\theta_*$, the DM model, $\theta_\star$, the mass-to-light ratio, $\Upsilon_v$, the (variable length) knot vector, $\xi$, and the (variable number of) unknown coefficients, $a_i$, of $\sigma^2_{rr}$. This is a very computationally demanding problem mainly due to the variable number of parameters.

EAs have the advantage that they are robust in finding a global optimum solution as well as being very fast. They also do not require the fitness function to be a probability distribution function, so any particular condition we consider as physical (that is not easy to be described in terms of some probability density function) can be encoded. Furthermore, they can easily treat variable number of unknowns. Thus, by using as a fitness function a model selection criterion, like the Akaike Information Criterion (AICc; Sugiura 1978; Hurvich & Tsai 1991), we can effectively fit and select the best model at each iteration.

3.1.2 Solution representation

In this section, we describe how we represent the solution inside the EA. Each candidate solution (termed individual in the EA framework) consists of two parts: the first is a binary string of variable length that represents the variable length knot vector, $\xi$. The second is a vector of real variables, $\theta$, of fixed length that represents the stellar and DM model parameters, as well as the smoothing penalty coefficient $q$ and constant mass-to-light ratio, $\Upsilon_v$. Therefore, an individual is represented as a tensor product of the two different types of variables, $\xi \otimes \theta$.

The binary string that represents the knot vector, $\xi$, has length $N_{\text{knots}} \times N_{\text{bits}}$, where $N_{\text{knots}}$ is the total number of individual knots and $N_{\text{bits}}$ is the length of the binary string that represents a single knot, $\xi_i$. For $N_{\text{bits}} = 6$ bits\textsuperscript{10}, we have 127 subdivisions of the real interval $[0, r]$. This is a very good approximation for the quality of data we have in hand. The large uncertainties of the observables make the need for a further refinement unnecessary. Internally each of the knots, $\xi_i$, is evaluated in the interval $[0, 1]$. Then the vector of knots is multiplied by the tidal radius of the system, as estimated from the King model, and thus the knots take positions in the real space of variables. In the case where a different stellar profile is used, we define the tidal radius as the virial radius of the DM halo.

3.1.3 Recombination

In this section, we detail the process of recombination, i.e. the combination of solutions from the EA population between individuals that produce new solutions (offspring). Recombination consists of the operation of crossover, where individuals exchange ‘genetic material’ (they combine their representation values), and mutation, where a single individual has some of its variables randomly changed.

As mentioned previously, each individual representation consists of two types of variables, a binary string for the representation of knots, $\xi$, and a vector of reals for all the model parameters, $\theta$, i.e. $\xi \otimes \theta$. Recombination takes place between variables of the

\textsuperscript{10} The first and last knots are fixed to real values of 0 and 1, i.e. $\xi_1 = 000000$, $\xi_n = 111111$. 

Figure 1. Flow diagram of the JEAnS algorithm. It consists of three phases. In phase 1, the EA is used for the evaluation of optimum knot distribution and best mass model. In phase 2, we use information from ideal theoretical models in order to determine the optimum smoothing parameters. In phase 3, we use MCMC to estimate the marginalized distributions of the model parameters, keeping the knots fixed.
same type. For the real coded variables, θ, the crossover we use is PCX (Deb, Anand & Joshi 2002), and the mutation operator is the Makinen, Periaux and Toivanen Mutation (MPTM) operator (Makinen, Periaux & Toivanen 1999). For the binary string of knots, we had to devise our own crossover and mutation operators. We therefore detail below the operation of recombination only for the knot vectors ξ between two individuals.

3.1.4 Knot crossover

We want the crossover operation between two knots to have the following characteristics: combine information from the total number and the value of the knots for each individual. Therefore, we perform a modified version of the two-point binary crossover: We select two segments of binary strings from each individual, such that the total length of each segment corresponds to an integer multiple of a single knot, ʾξ𝑖, of length ʾNbits. Then, these segments are exchanged and the resulting binary strings are sorted in ascending order. For the case where the size of the recombined individuals is 3,11 i.e. ʾξ = (ʾξ1, ʾξ2, ʾξ3) = (0, ʾξ1, 1), the crossover is a standard two-point crossover between the middle knots with the restriction the length of the substrings exchanged to be equal. With these operations, the total number of knots remains invariant, i.e. the crossover operation cannot modify the total number of knots. The crossover operation between two knots is summarized in Algorithm 1. A schematic representation of the knot crossover is presented in Fig. 2.

Algorithm 1 Crossover operator for knots ʾξ

Require: Parents ʾχ1, ʾχ2
Ensure: Offspring ʾχ
1: if dim1 = dim2 = 3 then
2: ʾ(ʾχ1, ʾχ2) ← (ʾχ1, ʾχ2) ▷ standard two point crossover
3: function REPAIR(ʾχ1)  
4:  end function
5: ʾSort(ʾχ1)  
6: function REPAIR(ʾχ2)  
7: ʾSort(ʾχ2)  
8: end function
9: else
10: Calculate random ʾn1, ʾn2 ∈ [1, ʾN1]  
11: Set {ʾn1, ʾn2} = [min(ʾn1, ʾn2), max(ʾn1, ʾn2)]  
12: Calculate integer ʾa = (ʾn2 − ʾn1)/ʾNbits  
13: Set (ʾn2 = ʾa1)  
14: function COPY((ʾn1, ʾn2))  
15: Copy bits from ʾχ1 between [ʾn1, ʾn2]  
16: end function
17: Repeat the same for ʾχ2
18: ʾ(ʾχ1, ʾχ2) ← (ʾχ1, ʾχ2)  
19: function REPAIR(ʾχ1)  
20: ʾSort(ʾχ1)  
21: end function
22: function REPAIR(ʾχ2)  
23: ʾSort(ʾχ2)  
24: end function
25: ʾχ ← best(ʾχ1, ʾχ2)  
26: end if

11 Recall that the first and last knots are held fixed to ʾξ1 = 0, ʾξm = 1.

3.1.5 Knot mutation

In order to get diversity in the knots, we want the mutation to be able to change both the number of knots and their value. Therefore, we split the mutation operator in two operations, each taking place with probability ʾp = 0.5: half of the times the mutation will modify only the number of knots, the other half only their value. When we modify the number of knots, we decide if we will delete or add a random knot according to a biased Bernoulli probability, flip(ʾpm), where ʾp0 = 0.005. If flip(ʾpm) = true we either add or delete a knot, at a random position, with probability ʾp = 0.5. When we want to modify the value of knots, we perform a standard binary flip bit mutation as described in Goldberg (1989), with probability flip(ʾpm). In this way, the mutation operator preserves diversity in both the number of knots and their value. We summarize the process in Algorithm 2. The variable ʾNbits corresponds to the total number of binary digits of the knot vector, excluding the first and last knots, which remain fixed to 0 and 1, respectively.

3.1.6 Evolution model

For the evolution of the EA, we used a modified version of the Generalized Generation Gap evolutionary model (hereafter G3; Deb et al. 2002). The modification is necessary in order to account for the two distinct types of variables in the individual representation, namely the variable length binary string, ʾξ, and the vector of reals, θ. We chose using the G3 evolutionary model because of its competitive performance over other evolution schemes (Hansen et al. 2010).

In the G3 algorithm, we create a temporary pool, ʾPµ−1, of parents consisting of ʾμ − 1 randomly chosen individuals. Each of the ʾμ − 1 parents is selected with equal probability from the population ʾP, excluding the best candidate, ʾˆχ. Then we create ʾλ offspring with the following scheme:

(i) Select a random individual, ʾχi, from the temporary pool, ʾPµ−1, with uniform probability.

(ii) Crossover the knot vector, ʾξi, of the best individual, ʾˆχ, with the knot vector, ʾξj, of ʾχi (this operation produces a single-knot vector).

(iii) Crossover the vector of reals, θ, of the best individual, ʾˆχ, with the vector of reals from the pool ʾPµ−1, according to the PCX crossover scheme (Deb et al. 2002). This operation uses the information from the whole pool of parents, ʾPµ−1, and produces one vector of reals centred close to the best candidate solution.

(iv) Mutate knots.

(v) Mutate θ according to the MPTM algorithm.

(vi) The offspring consists of the new knot vector and the new generated vector θ.

This process is repeated until ʾλ offspring is created. Then we select two candidates, ʾχ1 and ʾχ2, from the initial population, ʾP. We construct an augmented pool, ʾPµ+2, that consists of ʾλ offspring, ʾP0, and the ʾχ1 and ʾχ2. The two best candidates of this augmented pool, ʾPµ+2, replace the initial ʾχ1 and ʾχ2 in the population, ʾP. The modified G3 algorithm is summarized in Algorithm (3). This evolution method preserves the best candidate from the previous population if it remains the best solution after the recombination operation.

The algorithm set up is: total number of individuals in the population ʾNpop = 100, number of parents ʾμ = 5 and number of offspring ʾλ = 20. We run the algorithm for a sufficient number of generations until convergence (usually a few thousand generations). For the MPTM mutation, we used a constant probability of ʾp0 = 0.005. Unlike traditional genetic algorithms, in G3 the crossover operation
is always performed (crossover probability $p_c = 1$) and by design, it can jointly explore all parameter space and maintain diversity. We added mutation for additional diversity in the solution space.

Algorithm 2 Mutation operator

Require: $\chi$ Chromosome
Ensure: $\tilde{\chi}$ Mutated individual
1: Calculate random $p_1 = \text{unif}(0,1)$
2: if $p_1 \leq 0.5$ then $\triangleright$ Flip bits
3: for $i = 1 \ldots N_{\text{init}}$ do
4: if flip($p_m$) : true then $\triangleright$ Bernoulli biased
5: Flip bit
6: end if
7: end for
8: else $\triangleright$ Modify number of knots
9: if flip($p_m$) then $\triangleright$ Add/delete random knot
10: Calculate $p_2 = \text{unif}(0,1)$
11: if $p_2 \leq 0.5$ then
12: Insert a random bits knot
13: else
14: Delete a knot
15: end if
16: end if
17: end if

Algorithm 3 G3 evolution model

Require: Population $P$
Ensure: Updated population $P$
1: repeat
2: Select the best individual, $\tilde{\chi}$
3: Select $\mu - 1$ random individuals from $P$
4: $P_{\mu-1} = \{\chi_j, j = 1, \ldots, \mu - 1\}$
5: procedure Create($\lambda$ offsprng)$\tilde{\chi}$, $P_{\mu-1}$
6: $P_{\lambda} \leftarrow \{\}$ $\triangleright$ Empty pool
7: for $i = 1, \ldots, \lambda$ offspring do
8: random $\chi_i \in P_{\mu-1}$
9: $\chi_i \leftarrow \text{CROSSOVER}(\tilde{\chi}, \chi_i, P_{\mu-1})$
10: $\chi_i \leftarrow \text{MUTATE}(\chi_i)$
11: $P_{\lambda} \leftarrow \chi_i$
12: end for
13: return $P_{\lambda}$
14: end procedure
15: Select random $\chi_1, \chi_2 \in P$
16: $P_{\lambda+2} \leftarrow \{\chi_1, \chi_2\} \cup P_{\lambda}$
17: $P \ni (\chi_1, \chi_2) \leftarrow (\chi_1, \tilde{\chi}_2) \in (P_{\lambda+2})$
18: until Convergence

Figure 2. Crossover operation between the knots, $\xi$, of two individuals. For each individual, a region is selected at random that corresponds to an integer multiple of the number of bits that can represent a knot. Then these two regions are exchanged. The number of knots for each individual is not preserved; however, the total number of knots between both individuals is preserved.

3.1.7 Fitness function

The unknown parameters that fully specify a solution are the knots, $\xi$, the coefficients $a_i$, the stellar mass density parameters, $\theta_*$, the DM mass density parameters, $\theta_{\text{DM}}$, the mass-to-light ratio, $Y_V$, and the smoothing penalty coefficient, $q$. Let $\theta = \{\xi, \theta_*, \theta_{\text{DM}}, Y_V\}$ denote the subset of these that does not contain the unknown coefficients, $a_i$, and smoothing parameter, $q$. Due to the linearity of the equation (8) with respect to the $a_i$ coefficients, once we specify a set of values $\theta$ and $q$, we can find a unique solution for the set of unknowns, $a_i$. This follows from the form of the $\chi^2$ minimization of the penalty function to obtain the $a_i$:

$$-\ln \mathcal{L}_{\theta} = \sum_{i=1}^{N_{\text{data}}} \left[ \frac{(\sigma_i^2 - \sigma^2_{\text{los}}(R_i))}{2\sigma_i^2} \right]^2 + qW,$$  (17)

where $\sigma_i^2$ are the data values of the line-of-sight velocity dispersion, $\sigma^2_{\text{los}}; q$ and $W$ are given by equations (15) and (13) (detailed in Section 3.3). The functional form of $\ln \mathcal{L}_{\theta}$ is of convex type with respect to the $a_i$, unknowns,\(^\circ\) hence it has a unique solution.

Therefore, we break down the optimization problem to a hybrid one: a combination of heuristic search for the $\theta$ and $q$ parameters that are being treated by the EA and a convex minimization for the evaluation of the $a_i$. The convex minimization takes place inside the EA\(^\circ\) at each $\ln \mathcal{L}_{\theta}$ evaluation. In this way, we significantly speed up the convergence of the overall optimization. Furthermore, using convex minimization algorithms, we can easily encode meaningful restrictions to the unknown B-spline coefficients, $a_i$, in order to avoid non-physical solutions such as\(^\circ\) the smoothing penalty coefficient, $q$, and the smoothing penalty coefficient, $q$. Due to the linearity of the equation (8) with respect to the $a_i$ coefficients, once we specify a set of values $\theta$ and $q$, we can find a unique solution for the set of unknowns, $a_i$. This follows from the form of the $\chi^2$ minimization of the penalty function to obtain the $a_i$: 12

\[ f(\theta) = \frac{1}{1 + \text{AICc}(\theta)} \]  (18)

where

\[ \text{AICc} = -2 \ln \mathcal{L}_{\theta} + 2n + \frac{2n(n+1)}{N_{\text{data}} - n - 1} \]  (19)

and $n$ is the number of unknown parameters. AICc is the Akaike Information Criterion (AIC) with second-order bias correction (see

\(^\circ\) Least-squares minimization with a smoothing penalty is equivalent to quadratic programming minimization.

\(^\circ\) In practice, we also evaluate $q$ inside the EA, since specifying $\theta$ yields a unique solution for $a_i$ and $q$. This problem, however, is non-convex.

\(^\circ\) The restriction $q \geq 0$ we used in Diakogiannis et al. (2014a,b) is a stronger requirement than $q_{\text{los}} \geq 0$ but it cannot guarantee that $q_{\text{los}}^2 \geq 0$.

\(^\circ\) Source: http://www.qpOASES.org/
A novel JEAxS analysis of Fornax

3.2 Evaluation of optimum smoothing parameters

In this section, we describe how we calculate the value of the parameters \( \alpha, \beta \) of the hyperprior \( p(q|\alpha, \beta) \) in order to evaluate uncertainties in all of the model parameters through a MCMC process. The method we propose here is significantly different than what we presented in Diakogiannis et al. (2014b).

The hyperprior distribution \( p(q|\alpha, \beta) \) regulates the values selected for the \( q \) parameter from the MCMC algorithm that fits the model (now with fixed knots) to the data. Then we need a set of \( \{q_i\} \) data values in order to fit the \( p(q|\alpha, \beta) \) distribution to these and obtain estimates for the \( (\alpha, \beta) \) parameters.

Having evaluated a set of optimum model parameters, \( \theta \) with the use of the EA, we can use these as reference parameters (hereafter \( \theta^{\text{ref}} \)) to calculate synthetic data from some reference anisotropy profile, \( \beta^{\text{ref}}(r) \). We are free to choose any form of the anisotropy profile but we would like to use a profile that has similarities with the best knot vector solution from the EA. For example, we can use the generalized Osipkov–Merritt (Osipkov 1979; Merritt 1985) anisotropy profile

\[
\beta(r) = c_1 \frac{r^2}{r^2 + c_0^2}
\]

with \( c_0 \) equal to the mean value of the interior knots. Alternatively, we can also fit the system under investigation with the traditional method of Jeans modelling and use the highest likelihood fit as an ‘ideal theoretical model’, i.e. \( \theta^{\text{ref}}, \beta^{\text{ref}} \), from which we will draw conclusions for the smoothing hyperprior parameters, \( (\alpha, \beta) \).

For this reference profile, \( (\theta^{\text{ref}}, \beta^{\text{ref}}) \), we can estimate a reference B-spline approximation\(^\dagger\) of the radial velocity dispersion, \( \sigma^2_{\theta^{\text{ref}}}(r) = \sum_i a_i^{\text{ref}} B_i(r) \). The best smoothing parameter, \( q \), is the one whereby using \( \theta^{\text{ref}} \) as model parameters, and \( q \) as the smoothing parameter, the estimated solution, \( \hat{\alpha} \), has the closest proximity to the reference values, \( a_i^{\text{ref}} \). We use the following measure for the optimum penalty parameter \( q \):

\[
S(q) = \sum_{j=0}^{2} \sum_{i=1}^{n} |\Delta^j \hat{\alpha}_i(q) - \Delta^j a_i^{\text{ref}}|,
\]

where \( \hat{\alpha}_i(q) \) emphasizes the dependence of the solution values \( \hat{\alpha}_i \) on the smoothing parameter, \( q \). We estimate the coefficients \( \hat{\alpha}_i \) using the quadratic programming minimization of equation (17). The ‘differences’

\[
\Delta^0 a_i = a_i \\
\Delta^1 a_i = a_{i+1} - a_i \\
\Delta^2 a_i = a_{i+2} - 2a_{i+1} + a_i
\]

represent, respectively: the difference in the coefficients value, \( \Delta^0 \), the approximate slope, \( \Delta^1 \), and the approximate curvature, \( \Delta^2 \). That is, we require \( C^0, C^1 \) and \( C^2 \) proximity of the estimated solution to the reference profile. To get a robust set of values \( q \) that minimize equation (22), we create random realizations of synthetic data \( (R_i, \sigma^2_{\theta^{\text{ref}}}(R_i)) \), based on the reference profile. These random data sets have the same positions \( R_i \) as the true data set we wish to fit and are equal in number.\(^\dagger\) Then for each of these we minimize the function \( S(q) \), with respect to the \( q \) parameter, using some black box non-linear optimizer and obtain a value \( q \). Our choice of robust black box optimizer is the method Covariance Matrix Adaptation Evolutionary Strategy (CMA-ES; Hansen & Ostermeier 2001) and the C++ software implementation LIBCMAES.\(^\dagger\)

We summarize this process in Algorithm 4. Once we have a set of \( \{q_i\} \) data values, we fit the hyperprior function \( p(q|\alpha, \beta) \) to these and get estimates for the parameters \( (\alpha, \beta) \).

\(^\dagger\) We can do so by solving the equation

\[
\sigma^2_{\theta^{\text{ref}}}(r) = \sum_i a_i^{\text{ref}} B_i(r)
\]

for a large number of data values, \( \{r_i, \sigma^2_{\theta^{\text{ref}}}(r_i)\} \), with no smoothing penalty and standard linear algebra operations. We recommend linear or quadratic programming solvers, since these can incorporate physically plausible constraints, e.g. \( \sigma^2_{\theta} \geq 0 \).

\(^\dagger\) See Diakogiannis et al. (2014b) for more details.

\(18\) Source: https://github.com/beniz/libcmaes

Burnham & Anderson 2002, and references there in), and

\[
- \ln L_{\text{EA}} = \sum_{i=1}^{N_{\text{data}}} \left[ \frac{(\sigma^2_i - \sigma^2_{\theta^{\text{ref}}}(R_i))}{\delta \sigma^2_i} \right]^2 + \sum_{i=1}^{N_{\text{data}}} \left[ \frac{(\Sigma - \Sigma(R_i))}{\delta \Sigma_i} \right]^2 .
\]

AICc is the approximation of AIC for a small number of data compared to the number of unknowns. We used AICc instead of Bayesian Information Criterion since the former is more robust for small data sets while the latter is inappropriate when the number of parameters is comparable to the number of available data (Burnham & Anderson 2002). AICc is known to give the best model and solution according to the optimum trade-off between bias and variance (Hastie et al. 2001).

When we evolve the EA, we use \( n = n_{\text{coeffs}} \) in equation (19), i.e. \( n \) is the total number of unknown coefficients, \( a_i \), of the B-spline approximation \( \sigma^2_i \), not the total number of model parameters, \( n_{\text{params}} \). This stems from the fact that we are interested to identify the smallest possible knot vector that gives a good fit. Once we have established a good fit and we want to compare models, we calculate the AICc using all the model parameters. The reason we do this is related to multiobjective optimization (Zhou et al. 2011, see also Ghosh, Dehuri & Ghosh 2008): the AICc has two goals, to minimize the number of unknowns, \( n \), and find the best fit by minimizing \( -2 \ln L_{\text{EA}} \). So in principle, we have a multiobjective (i.e. vector) optimization problem of the form \( f = (-2 \ln L_{\text{EA}}, n) \) of two conflicting goals, where \( f \) is a two-dimensional fitness function. Combining both information criteria into one scalar function, i.e. the AICc, gives just one solution instead of the whole Pareto front (Coello, Lamont & Veldhuizen 2006) of equally good solutions. This also means that the optimization is sensitive to the value of the parameters, \( \alpha, \beta \) in the AICc by using the total number of parameters, \( n_{\text{params}} \), instead of the total number of model parameters, \( n_{\text{coeffs}} \). The best kinematic profile. After the EA optimization, by using the total number of parameters \( n_{\text{params}} \) in the AICc, we can perform model inference between competing mass models.

\(16\) Source: https://github.com/beniz/libcmaes
3.3 Evaluation of marginalized distributions for the model parameters

Once we have selected an optimum knot distribution, we use Bayesian inference in order to evaluate uncertainties of the various model parameters. For the evaluation of the marginalized distributions for our model parameters, we used an MCMC approach for the full likelihood in equation (11). Specifically, we designed a stepper that is a combination of Differential Evolution MCMC (Braak 2006) and affine invariance MCMC (Goodman & Weare 2010) in a parallel tempering scheme. The library we used is POPMCMC++, a C++ library for population MCMC. It can be downloaded from https://github.com/feevos/poppmcmc.

Since it is difficult to add hard bound constraints in the distribution function when performing MCMC sampling, the user of our method should carefully process the chains, after sampling, by removing possible model parameters that violate physical constrains (e.g. $\sigma_n^2 < 0$).

4 EXAMPLE

In this section we will present an example of the usage of our algorithm. Initially, the EA will evaluate the optimum knot distribution, $\xi$, as well as the best candidate DM model. For a robust modelling, we should also try various stellar models; however, for simplicity, we restrict our analysis to varying only the DM models. Once we have the optimum DM knot distribution, $\xi^{EA}$, and the best candidate model of DM, we use MCMC to derive uncertainties in the model parameters. For the evaluation of the marginalized distributions for our model parameters, we used an MCMC approach for the full likelihood in equation (11). Specifically, we designed a stepper that is a combination of Differential Evolution MCMC (Braak 2006) and affine invariance MCMC (Goodman & Weare 2010) in a parallel tempering scheme. The library we used is POPMCMC++, a C++ library for population MCMC. It can be downloaded from https://github.com/feevos/poppmcmc.

Since it is difficult to add hard bound constraints in the distribution function when performing MCMC sampling, the user of our method should carefully process the chains, after sampling, by removing possible model parameters that violate physical constrains (e.g. $\sigma_n^2 < 0$).

4.1 Synthetic data

We constructed a set of synthetic data points from a King (1966) brightness profile, a Burkert (1995) DM mass density, and an MLT anisotropy profile (Tiret et al. 2007, see also Mamon et al. 2013) defined by

$$\beta_{\text{MLT}} = \beta_{\infty} \frac{r}{r + r_{\beta}}.$$ (23)

Our choice of parameters that produce a relatively difficult profile that is not approximated well with a uniform knot distribution is

$$\theta^\text{ref} = \{w_0^\text{ref} = 7.5, r_{\text{ref}}^\text{ref} = 50 (\text{M}_\odot \text{ pc}^{-3}), r_{c}^\text{ref} = 100 (\text{pc})\}$$
$$\theta^\text{rand} = \{w_0^\text{rand} = 10 (\text{M}_\odot \text{ pc}^{-3}), r_{c}^\text{rand} = 200 (\text{pc})\}$$
$$\theta_{\beta} = \{\beta_{\infty} = -0.5, r_{\beta} = 15 (\text{pc})\}.$$
generations of the EA in log space. In the top panel, we plot the fitness value, $f$, as a function of the generations, $N_{\text{gen}}$. Solid lines correspond to the maximum fitness value of the population, $f_{\text{max}}$, while dotted lines to the average fitness, $\langle f \rangle$, of the generation. As the generations evolve, the average fitness value approaches the fitness of the best individual. This is an indication of the convergence of the algorithm around the best solution. In the middle panel, we plot the standard deviation of the fitness values of the population. The standard deviation, $\sigma_f$, initially rises, i.e. the spread of the fitness values increases. This is because the EA tries to encapsulate in its population the best solution; therefore, it spreads the values of the solutions in order to explore parameter space. As the EA progresses and the best candidate has been found and is included in the population, the population tends to ‘shrink’ around the best solution. This is why the average value gets closer to the highest fitness value, and the standard deviation gets minimized. Some spikes that are observed in $\sigma_f$ are because of recombination: a better solution was found away from the median of the population and the EA spreads until it encapsulates it. For example, the sudden change in fitness and AICc of the BH model at generation $\approx 2000$ is accompanied by a spike in the dispersion of the fitness.

In the bottom panel, we plot the corresponding values of the AICc information criterion. Note that this AICc is evaluated from equation (19) with $n$ being the total number of parameters, i.e. $n = \dim(\theta_i, \theta_{\text{DM}}, \{a_i\})$. This is why the highest fitness value does not necessarily correspond to the minimum AICc. In Table 1, we report the AICc values, as well as the differences, $\Delta$, between the best AICc model (Burkert) with its competitors. For selecting the most probable model, we follow Burnham & Anderson (2002), who state that model selection between two competing models is conclusive if the difference of their AICc values is greater than 10. The Burkert DM model (the reference from which the synthetic data were created) is clearly distinguished from competing solutions, so we can use them. Our estimated parameter values are close except for the $\rho_{\text{DM}}$ and the DM model parameters. We cannot make any prediction at this stage of the deviation from the true values since we have no uncertainty for our estimates. After all, we are interested only in selecting the best candidate mass model, as well as identifying a really good knot distribution. Optimum smoothing based on ideal theoretical models results in reliable estimates of the kinematic profiles after we have selected a good knot distribution.

The EA-recovered parameters are $\xi^\text{EA} = \{0, 0.21, 1\}$ and $\{w_0^\text{EA} = 6.19, \rho_{\text{EA}} = 51.46 (M_\odot \text{ pc}^{-3}), r_{\text{EA}}^\text{c} = 107.13 (\text{pc}), \rho_{\text{EA}}^* = 27.72 (M_\odot \text{ pc}^{-3}), r_{\text{EA}}^* = 105.04 (\text{pc}), q^\text{EA} = 0.52\}$. The knots $\xi^\text{EA}$ are evaluated in the unit interval and must be scaled to $[0, r_i]$ before we can use them. Our estimated parameter values are close except for the $w_0^\text{EA}$ and the DM model parameters. We cannot make any prediction at this stage of the deviation from the true values since we have no uncertainty for our estimates. After all, we are interested only in selecting the best candidate mass model, as well as identifying a really good knot distribution that will allow us to perform a robust statistical fit.

Table 1. Synthetic data, AICc comparison of models. A value of $\Delta > 10$ states model selection conclusive (Burnham & Anderson 2002).

| Model      | AICc   | $\Delta$ |
|------------|--------|----------|
| King, Burkert | 91.76  | 0.0      |
| King, NFW   | 158.15 | 66.38    |
| King, Einasto| 170.72 | 78.96    |
| King, BH    | 190.48 | 98.72    |

Figure 4. Evolution of most favoured model for the synthetic data set. Top panel: maximum fitness value [solid line – equations (18) and (19)], average fitness (dash-dotted line) and fitness variance (dashed line). Middle panel: evolution of the average value, $\langle \theta_i \rangle$, of the model parameters. Bottom panel: evolution of the variance, $\sigma_{\theta_i}$, of each of the model parameters.

4.2.2 Confidence interval for model parameters

Once we have chosen a suitable knot vector, $\xi$, we can focus on repeating the fit with MCMC, keeping the knots, $\xi_i$, fixed so as to
4.2.3 Evaluation of $\alpha$, $\beta$ parameters

Our goal is to estimate the $\alpha$, $\beta$ parameters of the hyperprior $p(\tilde{q}(\alpha, \beta))$. In order to do so, we need to estimate the parameters $\alpha$, $\beta$ of the hyperprior $p(\tilde{q}(\alpha, \beta))$ that regulate the smoothing. For this, we apply Algorithm 4 (see Section 3.2).

4.2.4 Deriving uncertainties on the model parameters

In order to do so, we first need to estimate the parameters $\alpha$, $\beta$ of the hyperprior $p(\tilde{q}(\alpha, \beta))$ that regulate the smoothing. For this, we apply Algorithm 4 (see Section 3.2).}

Once we have the maximum likelihood values, $\hat{\alpha}$, $\hat{\beta}$, we substitute them in equation (11) and with an MCMC we obtain the Markov Chains of the model parameters. We use the restriction that $\sigma_{\alpha}(r_t) \approx \sigma_{\beta}(r_t) \approx \sigma_{\tilde{q}}(r_t) \approx 0$. From an empirical point of view, close to the tidal radius, for bound stars, the radial velocity must approach zero ($r_t$ is a turning point for all bound particles approaching it). Hence, we expect that the uncertainty in $v_r$ values is very small; this should lead to $\sigma_{\alpha}(r_t) \to 0$. On the other hand, if we assume that there are purely tangential motions, these must have very small velocity in order to remain in bound orbits (due to the small value of the gravitational force at $r \approx r_t$), i.e. $v_t \approx 0$. Hence, we expect again that $\sigma_{\beta}(r_t) \to 0$. See also Merritt (2013) for a theoretical proof that as $r \to \infty$ this restriction is equivalent to the virial theorem for a self-gravitating system in equilibrium. For the case where we wish to use a stellar model different from the King profile, the tidal radius can be evaluated from the virial radius of the system, i.e. $r_t \approx r_{vir}$. Our restriction on the various velocity dispersions at $r_t = r_{vir}$ remains a good approximation.

In Fig. 7, we plot the Markov Chains of the model parameters as obtained from the MCMC. In all marginalized distributions of the Markov chains, the reference values of the profile that created the synthetic data set are captured. There is a strong correlation between the $\rho_{\psi}$, $\rho_{\phi}$, and $\rho_{\psi\phi}$ parameters and this is one reason why the evolutionary algorithm gives the greatest error in the estimation of these values. This may have also affected the $w^\psi_{\alpha \beta}$-evaluated parameter that lies outside the marginalized distribution of the MCMC chains. Since the EA uses AICc for the best model fit, it cannot account for the optimum smoothing from mock stellar systems that MCMC has built in its prior information. This is a source of bias in the EA estimates, but in all of our tests with mock data sets, it does not seem to affect the model selection.

In Fig. 8, we plot the fit of the various velocity moments. In the top panel, we see the line-of-sight velocity dispersion, $\sigma_{\psi\phi}$, the true reference profile that corresponds to the synthetic data set, as well as the synthetic data. In the middle panel, we plot the maximum likelihood radial velocity dispersion, $\sigma_{\beta}$, and the corresponding reference value. We also show the control polygon that regulates the geometric shape of the B-spline $\sigma_{\beta}$ function. Observe that there are two control points with r-coordinate closer to the origin, $r = 0$, something evaluated by the EA. In the bottom panel, we plot the tangential velocity dispersion, $\sqrt{\sigma_{\psi}/2} = \sqrt{(\sigma_{\psi}^2 + \sigma_{\phi}^2)/2}$. In all
panels, the grey region corresponds to the $1\sigma$ uncertainty interval of the estimated values of all model parameters. The reconstruction is satisfactory. Even in the case where the fitted profile deviates from the true one (top and bottom panels for $R, r > 0.6 \text{kpc}$), the uncertainty is greater towards the true reference profile. Note that there is a difference in the best solution found by the EA (Fig. 5) and the MCMC (Fig. 8). The reason is that in the likelihood used in the MCMC, we have encoded the smoothing information from ideal mock systems. This information is not available for the EA runs.

In Fig. 9, we plot the estimated anisotropy profile, $\beta(r)$, as well as the true reference anisotropy, $\beta_{\text{MLT}}(r)$ (recall $r_t \approx 1 \text{kpc}$). The fact that we do not fit directly a functional form for $\beta(r)$ has several benefits and a penalty: the uncertainty in the $\beta(r)$ estimates is large, especially for $r > 0.6 \text{kpc}$. This is a result of the functional form that defines $\beta(r)$: the division of $\sigma_{\text{tt}}^2$ by $\sigma_{\text{rr}}^2$ propagates large errors, especially closer to the tidal radius, where both $\sigma_{\text{tt}}^2$ and $\sigma_{\text{rr}}^2$ tend to zero.

5 APPLICATION TO THE FORNAX DWARF SPHEROIDAL

In this section, we apply the JEAnS algorithm to the Fornax dSph and present our results. In Appendix B, we give some further insight on differences between EA adaptive knot modelling and modelling with a uniform knot distribution.
5.1 Data

Fornax is a dSph galaxy of the Local Group. It appears to be slightly flattened, with ellipticity $e \approx 0.3$ (Battaglia et al. 2015). It is at a distance $d = 138 \pm 8$ kpc (Mateo 1998) from Sun and the coordinates of its centre are RA: $2^h 40^m 04^s$, DEC: $-34^\circ 31^\prime 30^\prime$ (J2000; Walcher et al. 2003; Coleman et al. 2005). We summarize this information in Table 2. We used published heliocentric velocity values and membership characterization from Walker et al. (2009a,b). Targets were considered members if their membership probability was greater than 0.8. This resulted in total 2248 Fornax members. For the brightness, we constructed the projected number density profile, normalized to the total luminosity of Fornax, $L_V = (1.79 \pm 0.69) \times 10^7 L_\odot$ (Łokas 2009, based on Irwin & Hatzidimitriou 1995 and Mateo 1998). We did not account for the error in the $L_V$ value, instead the errors were estimated using the bootstrap method. In order to estimate the velocity dispersion values, we binned the data in 19 variable size circular annuli bins, each one containing 82 stars. We estimated the line-of-sight velocity dispersions in each radial bin via our MCMC scheme (assuming again Gaussian line-of-sight velocity distributions). In Fig. 10, we plot the positions of the Fornax members on the tangent plane (left-hand panel) and the heliocentric line-of-sight velocities with respect to the distance from the cluster centre (right-hand panel).

5.2 Results

We model Fornax by assuming a King (1966) brightness profile, a constant V-band mass-to-light ratio, $\Upsilon_V$, and in addition the following separate DM components: NFW, Burkert, Einasto and a BH in the centre of the galaxy. In all of the models, there is a constant mass-to-light ratio, $\Upsilon_V$, but only in one do we not use a separate DM mass profile; thus, in total we have five different mass models. We refer to the model with no separate DM component as const$\Upsilon_V$. In Table 3, we give the range of the model parameters that we used in the evolutionary algorithm. Clearly, our range of parameters range from unrealistically small to unrealistically large mass models. Despite the elliptical shape of Fornax, we make the simplifying assumption that it is a spherically symmetric object in virial equilibrium that is not affected by the tidal field of the host. This seems a viable scenario (Łokas 2009; Battaglia et al. 2015). In our analysis, we do not account for contamination from binaries; however, numerical simulations demonstrate that this should not have a significant effect in our dispersion estimations (Jardel & Gebhardt 2012, and references therein).

We start by running the EA for the five different models. The set up for the EA is the same as in the example with the mock data set (G3 evolution model, $\mu = 5, \lambda = 20, p_m = 0.005$, see Section 3.1.6). We evolved the algorithm for 3500 generations; this was adequate for convergence. In Fig. 11, we plot (top panel) the evolution of the maximum fitness value (solid lines), as well as the average fitness for the population (dotted lines). As the algorithm converges around the global maximum, the average value, $\langle f \rangle$, converges to the maximum fitness, $f_{\text{max}}$. In the middle panel, we show the evolution of the dispersion in the fitness values, $\sigma_f$. After approximately 100 generations, the algorithm has estimated a global maximum and from that point and on the dispersions decreases gradually. Finally in the bottom panel we plot the AICc. We emphasize that for the evaluation of the AICc in this panel, we used all of the model parameters, not just the number of unknown coefficients, $a_i$.

The simple model with constant mass-to-light ratio and no separate DM component is favoured (AICc $= 86.50$). The second best is the one with a BH in the centre (AICc $= 89.49$). For model inference with the AICc criterion, one is interested in the difference of the AICc values with the best model (lowest value of AICc). In Fig. 12, we visualize these differences. The vertical axis is in log scale. In Table 4, we list all values of AICc as well as the differences $A^\prime = \text{AICc} - \min(\text{AICc})$. Model comparison is

---

**Table 2. Fornax parameters (Łokas 2009).**

| Centre (J2000) | $L_V(10^7 L_\odot)$ | Distance (kpc) |
|----------------|---------------------|----------------|
| RA: $2^h 40^m 04^s$ | 1.79 ± 0.69 | $d = 138 \pm 8$ |
| DEC: $-34^\circ 31^\prime 30^\prime$ | | |

---

**Figure 8.** MCMC highest likelihood fit (blue curves) of second-order velocity moments for the synthetic data set. The grey region corresponds to $1\sigma$ uncertainty region in all model parameters. The dashed lines correspond to the true values of the reference profile from which the synthetic data were created. Black squares are the control points of the control polygon (solid black line) – see Diakogiannis et al. (2014a) for definition and details.

**Figure 9.** As Fig. 8, but for the anisotropy profile $\beta(r) = 1 - \sigma_r^2/(2\sigma_\theta^2)$ for the synthetic data set.
A novel JEA NS analysis of Fornax

Table 3. Range of parameters for the various models we used inside the evolutionary algorithm for the case of Fornax.

| Model     | Parameter       | $\theta$ range                  |
|-----------|-----------------|---------------------------------|
| King      | $W_0$           | [0.5, 20]                       |
|           | $\rho_0$        | [10^{-3}, 0.4] M pc^{-3}        |
|           | $r_c$           | [100, 1500] pc                  |
| Burkert + NFW | $\rho_0$        | [0.0, 0.4] M pc^{-3}           |
|           | $r_s$           | [100.0, 2000.0] pc              |
| Einasto   | $\rho_e$        | [0.0, 2.0] M pc^{-3}           |
|           | $r_e$           | [2.0, 2000.0] pc                |
|           | $n$             | [1, 10]                         |
| BH        | $M_*$           | [10^2, 10^6] M pc               |
| Mass-to-light ratio | $\Upsilon_V$ | [0.2, 100] M pc / L pc          |

Table 3. Range of parameters for the various models we used inside the evolutionary algorithm for the case of Fornax.

conclusive (Burnham & Anderson 2002): the best model is a simple King profile with constant mass-to-light ratio. There is no need for the assumption of a separate DM component with a significantly different mass profile. For the best model, the knot distribution estimated from the EA is uniform with only three knots, i.e. $\xi = [0.00, 0.50, 1.00]$. Remarkably, despite the apparent flattening of Fornax that should increase the dispersion profile and perhaps suggest a different than the simple mass-follows-light DM component, our algorithm rejects such a scenario. The rejection of a cuspy profile is in accordance with the results of Amorisco et al. (2013) and Walker & Peharrubia (2011). We estimate a total mass, $M_{\text{tot}} = 1.613^{+0.058}_{-0.075} \times 10^8 M_\odot$, and a mass-to-light ratio, $\Upsilon_V = 8.93^{+0.52}_{-0.47} M_\odot/L_\odot$. The best-fitting King profile has parameters $w_0 = 2.60^{+0.12}_{-0.10}, \rho_0 = 0.120^{+0.009}_{-0.004} M_\odot pc^{-3}, r_c = 685^{+29}_{-18}$ pc. We summarize our results in Table 5. In Fig. 13, we plot (left-hand panel) the marginalized distributions of mass-to-light ratio and total mass. On the right-hand panel, we plot the normalized histogram of the marginalized total mass distribution. In Table 6, we list the best estimates for the knots and parameters from the Evolutionary Algorithm. The knots are divided by the tidal radius for each system and scaled in the interval [0, 1]. It needs to be emphasized that the tidal radius of each of the models depends on the adopted DM profile. This results from the solution of the Poisson equation for the King brightness profile. In all cases, the tidal radius of the system, $r_t$, is found to be much larger than the outermost radial bin of data.

Figure 11. EA optimization for various DM competing models for the Fornax dSph. Top panel: fitness value [equations (18) and (19)] of the best individual (solid line) and average value of fitness (dotted line). Middle panel: standard deviation of the fitness values of the population. Bottom panel: AICc criterion of the competing models.

In Fig. 14, we plot in the top left panel the brightness fit. In the bottom left panel, bottom left panel, we plot the line of sight, $\sigma_{\text{los}}$, we estimated as well as the maximum likelihood fit for the best model (dashed line). In the top right panel, we plot the radial
and radial motions are equivalent, thus in accordance with theoretical predictions: clearly, at the bottom right panel, the tangential, $\beta$-olation and should not be trusted. In general, as stated previously, increasing. The vertical dashed line marks approximately the point $\beta \rightarrow 0$ for cored stellar profiles, $\rho_c$, and a non-singular potential in the origin $\lim_{r \rightarrow 0} \rho(r) = 0$.

In Fig. 15, we plot the anisotropy profile. This starts from zero, in accordance with theoretical predictions: clearly, at $r = 0$ tangential and radial motions are equivalent, thus $\beta(0) = 0$. The anisotropy becomes progressively negative, with the uncertainty in the fit increasing. The vertical dashed line marks approximately the point where the range of our data ends, so anything beyond that is extrapolation and should not be trusted. In general, as stated previously, our algorithm does not constrain well the velocity anisotropy, $\beta(r)$.

Figure 13. Mass-to-light ratio, $\mathcal{Y}_V$, and total mass, $M_{\text{tot}}$, as estimated from the MCMC chains for the Fornax dSph.

Table 6. EA estimated parameters for the various models we used for the case of Fornax. The mass density constants $\rho_{\bullet}$, are evaluated in $M_\odot$ pc$^{-3}$.

| Model      | Parameter | Value                  |
|------------|-----------|------------------------|
| Const $\mathcal{Y}_V$ | $\xi$ | $\{0, 0.500, 1\}$ |
|            | $\theta_\star$ | $\{w_0, \rho_{\bullet}, r_c\}$ | $\{2.537, 0.121, 691.613\}$ |
|            | $\mathcal{Y}_V$ | $8.968 M_\odot / L_\odot$ |
|            | $\theta_\star$ | $\{M_\bullet\}$ | $7.0565 \times 10^7 M_\odot$ |
| BH         | $\xi$ | $\{0, 0.496, 1\}$ |
|            | $\theta_\star$ | $\{w_0, \rho_{\bullet}, r_c\}$ | $\{3.535, 0.388, 380.678\}$ |
| NFW        | $\xi$ | $\{0, 0.125, 1\}$ |
|            | $\theta_\star$ | $\{w_0, \rho_{\bullet}, r_c\}$ | $\{1.809, 0.173, 761.048\}$ |
|            | $\mathcal{Y}_V$ | $6.816 M_\odot / L_\odot$ |
|            | $\theta_\star$ | $\{\rho_{\bullet}, r_c\}$ | $\{0.311, 170.067\}$ |
| Burkert    | $\xi$ | $\{0, 0.123, 1\}$ |
|            | $\theta_\star$ | $\{w_0, \rho_{\bullet}, r_c\}$ | $\{2.551, 0.126, 687.322\}$ |
|            | $\mathcal{Y}_V$ | $8.594 M_\odot / L_\odot$ |
|            | $\theta_\star$ | $\{\rho_{\bullet}, r_c\}$ | $\{0.042, 196.315\}$ |
| Einasto    | $\xi$ | $\{0, 0.131, 1\}$ |
|            | $\theta_\star$ | $\{w_0, \rho_{\bullet}, r_c\}$ | $\{2.263, 0.173, 1077.640\}$ |
|            | $\mathcal{Y}_V$ | $8.830 M_\odot / L_\odot$ |
|            | $\theta_\star$ | $\{\rho_{\bullet}, r_c, n\}$ | $\{0.007, 1324.556, 2.278\}$ |

velocity dispersion, $\sigma_r$, as this is recovered from our algorithm. In the bottom right panel, the tangential, $\sqrt{\sigma_r^2 / 2}$, component is shown. In all figures, the grey region corresponds to $1\sigma$ uncertainty in all parameters.

In Fig. 15, we plot the anisotropy profile. This starts from zero, in accordance with theoretical predictions: clearly, at $r = 0$ tangential and radial motions are equivalent, thus $\beta(0) = 0$. The anisotropy becomes progressively negative, with the uncertainty in the fit increasing. The vertical dashed line marks approximately the point where the range of our data ends, so anything beyond that is extrapolation and should not be trusted. In general, as stated previously, our algorithm does not constrain well the velocity anisotropy, $\beta(r)$.

21 This can also be verified from the Jeans equation, equation (4) in the limit $r \rightarrow 0$ for cored stellar profiles, $\rho_\star$, and a non-singular potential in the origin $\lim_{r \rightarrow 0} \rho(r) = 0$.

22 The GDSAI states that $\beta \leq \gamma(r) / 2$, where $\gamma(r) = -d \log \rho_\star(r) / d \log (r)$. 

Figure 12. Comparison of the AICc convergence criterion for the Fornax dSph. The differences $\Delta^* = \text{AICc} - \min \text{(AICc)}$ are plotted. The case of $\text{const} \mathcal{Y}_V$ (simple King profile with constant mass-to-light ratio and no separate DM component) is clearly favoured over all competing models ($\Delta > 10$).

Table 4. Fornax AICc comparison of models.

| Model      | AICc | $\Delta$ |
|------------|------|----------|
| King, $\mathcal{Y}_V$ | 86.50 | 0.0 |
| King, $\mathcal{Y}_V$, BH | 98.49 | 11.99 |
| King, $\mathcal{Y}_V$, NFW | 112.66 | 26.15 |
| King, $\mathcal{Y}_V$, Burkert | 114.36 | 27.85 |
| King, $\mathcal{Y}_V$, Einasto | 135.38 | 48.88 |

Table 5. MCMC estimated parameters for the best model for the case of Fornax. The knots are scaled in the interval $[0, 1]$.

| Best model | Parameter | Value |
|------------|-----------|-------|
| Const $\mathcal{Y}_V$ | $\xi$ | $\{0, 0.500, 1\}$ |
|            | $w_0$ | $2.60 \pm 0.14$ |
|            | $\rho_{\bullet}$ | $0.120 \pm 0.009$ $M_\odot$ pc$^{-3}$ |
|            | $r_c$ | $685 \pm 38$ pc |
|            | $\mathcal{Y}_V$ | $8.93 \pm 0.32$ $M_\odot / L_\odot$ |
|            | $M_{\text{tot}}$ | $1.61 \pm 0.050 \times 10^4 M_\odot$ |
|            | $\langle \beta \rangle$ | $-0.95^{+0.78}_{-0.72}$ |
A novel JEAxS analysis of Fornax

Figure 14. Surface brightness and kinematic fits for the Fornax dSph. Top left panel: highest likelihood surface brightness fit and data values. Bottom left panel: line-of-sight velocity dispersion values. Blue points correspond to observed values. The dotted purple line is the highest likelihood fit. Top right panel: radial velocity dispersion fit. Bottom right panel: tangential velocity dispersion, $\sigma_{t}^2 = \sigma_{\phi t}^2 + \sigma_{\theta t}^2$. In all panels, the grey region corresponds to the 1σ uncertainty interval in all model parameters.

Figure 15. Velocity anisotropy profile of the Fornax dSph. Grey region corresponds to 1σ uncertainty interval in all model parameters. The vertical dashed line marks the end point of our data, i.e. the limit of trusted predictions. Beyond this point, the profile is extrapolated and should not be trusted. The horizontal dash-dotted line corresponds to the physically limiting value $\beta = 1$. We also overplot the maximum value of the anisotropy, $\beta_{\text{max}}^{\text{GDSAI}}$ (solid black line), as predicted according to the GDSAI.

Finally, in order to understand why the EA chose to reject the models with a separate DM component, we plot in Fig. 16 the $\sigma_{\text{los}}$ profiles for the best EA-selected competing models. All of the models with a DM component have a sharp turn close to the origin. This is the reason the EA rejects them: they possess unnecessary complexity. In contrast, the const$\Upsilon_{V}$ as well as the BH model has much smoother profiles close to the origin.

5.2.1 Comparison with previous work

Walker et al. (2007) perform a simple NFW profile fit, with the assumption of constant anisotropy. They estimate a total mass of $M_{\text{vir}} \sim 10^9 M_\odot$, while the total mass within the maximum data point ($r_{\text{max}}$) is $M_{\text{max}} = 1.8 \times 10^8 M_\odot$. Using $r_{\text{max}} \approx 1.8$ (kpc), we estimate $M_{\text{max}} \approx 1.549 \times 10^8 M_\odot$. While we are close to their $M_{\text{max}}$, we differ by orders of magnitude to their total (virial) mass estimate. The authors fit a simple King (isotropic) profile with a constant mass-to-light ratio, and they exclude it since it clearly fails to recover flat dispersion profiles. Quite remarkably, when we use a simple King profile, but with arbitrary anisotropy, we are able to recover the relatively flat line-of-sight velocity dispersion profile of Fornax. So a simple mass-follows-light model, with a non-parametric anisotropic profile, is a viable candidate (Fig. 14).

This result is also in accordance with the suggestions of Battaglia et al. (2015) that ‘baryons and DM may contribute similarly to the mass budget in Fornax’s central regions’.

Klimentowski et al. (2007) modelled Fornax by assuming a simple mass-follows-light system, a constant anisotropy and by using the method of interlopers removal (den Hartog & Katgert 1996). The authors estimated an anisotropy $\beta = -0.17^{+0.37}_{-0.63}$, a mass-to-light ratio $M/L = 11.3^{+2.1}_{-1.8} M_\odot/L_\odot$ and mass $M = 2.1 \times 10^8 M_\odot$. They also performed a simple $2.58\sigma$ cut in the line-of-sight velocities and again modelled the dwarf, finding a more tangential anisotropy, $\beta = -1.82^{+1.02}_{-2.66}$ and a smaller mass-to-light ratio $M/L = 10.6^{+1.8}_{-1.7} M_\odot/L_\odot$. The anisotropy and mass-to-light ratios are, within error bars, in close proximity to our results and their mass estimate is of the same order of magnitude with our estimates.

Figure 16. Best EA $\sigma_{\text{los}}$ profiles for all competing models for the Fornax dSph.
Lokas (2009) modelled Fornax by assuming a simple mass-follows-light model, a constant anisotropy and using information from the kurtosis of the line-of-sight velocities. They estimated a total mass $M_{\text{tot}} = (1.57 \pm 0.07) \times 10^8 M_{\odot}$, which is in close proximity with our estimates. Similar results hold also for her constant mass-to-light ratio, $\Upsilon_V$, which was estimated to be $\Upsilon_V = (8.8 \pm 3.8) M_{\odot}/L_{\odot}$. Our predictions agree within the error bars; however, our mass-to-light ratio is more tightly constrained. Furthermore, she also predicted a tangential anisotropy but in her case it is constant $\beta = -0.3 \pm 0.15 < 0$. In her work, the author does not perform model inference between competing mass models.

Amorisco et al. (2013) used a multiple stellar population model. They estimated a scale radius for a cored (Burkert) DM profile of $3.8\,r_c$. They base their model selection on a logarithmic one with and without a BH. They do not include in their method for modelling Fornax. They test various DM profiles and the use of different tracer profiles; this is currently a work in progress. Some of our goals for our future work is to expand our method with outputs from numerical simulations (Diakogiannis et al. 2014b).

Jardel & Gebhardt (2012) model Fornax using the Schwarzschild (1979) method. They test three DM models, a NFW profile and a core profiles. They estimate a mass within 1 kpc radius of $M_{1\,\text{kpc}} \approx 1.13 \times 10^8 M_{\odot}$. They base their model selection on $\chi^2$ comparison (which is a weak model inference method) and conclude that a cored model without a BH is a better description of the dSph.

Breddels & Helmi (2013) also use the Schwarzschild (1979) method for modelling Fornax. They test various DM profiles and calculate the Bayesian evidence for the most probable model. Their model selection does not give a clear distinction between cuspy and core profiles. They estimate a mass within 1 kpc radius of $M_{1\,\text{kpc}} \approx 10^8 M_{\odot}$, which is in close proximity with our estimate at the same distance, $M_{1\,\text{kpc}}(r = 1\,\text{kpc}) \approx 1.09 \times 10^8 M_{\odot}$. They estimate an average constant anisotropy, $(\beta \sim -0.2 \pm 0.2$. This range is consistent with the average anisotropy we calculate.

6 CONCLUSIONS

In this section, we report our conclusions from this contribution. For the convenience of the readers, we separate them in conclusions for the JEANS algorithm we developed and for the case of the Fornax dSph.

6.1 JEANS modelling

Building on previous work (Diakogiannis et al. 2014a,b), we improve our method of modelling spherical systems independent of velocity anisotropy assumptions. For this, we develop an evolutionary algorithm that evaluates the optimum knot distribution for the B-spline representation of $\sigma_{\beta}^2$, i.e. the best kinematic profile independent of the anisotropy $\beta(r)$. In addition, the JEANS discriminates between competing mass models using the AICc model selection criterion. This model inference criterion has the advantage that is fast to evaluate and it can be applied to small samples of data. It has the disadvantage that it does not include information from the whole range of Markov chains in the MCMC. As a result, it is not as robust as Bayesian inference methods, e.g. model inference using Bayesian evidence (Feroz, Hobson & Bridges 2009). In tests with synthetic data sets, our method performed well and managed to discriminate between competing DM models.

We extend the notion of empirical Bayes (Casella 1985) by presenting a new algorithm for the evaluation of prior information from ideal theoretical models for the optimum smoothing of $\sigma_{\beta}^2$ for the line-of-sight, $\sigma_{\beta}^2$, fit. This new version results in tighter constraints on the smoothing hyperprior parameters and is computationally much faster than the one presented in Diakogiannis et al. (2014b).

Finally, with our Jeans modelling independent of assumptions on velocity anisotropy, we are able to discriminate between different parametric mass models. Thus, we are able to model independent of the mass-anisotropy degeneracy.

6.2 Fornax

We apply our algorithm to Fornax dSph galaxy. Our method uses AICc for model inference. Based on the available brightness and kinematic data sets, our algorithm predicts conclusively ($\Delta(\text{AICc}) > 20$, Burnham & Anderson 2002) that there is no need for a separate DM component in the dwarf galaxy. That is, from a variety of cored and cuspy DM profiles and modelling independent of the MAD, our best candidate is a simple mass-follows-light model. This does not imply that there is no DM in the dSph; however, it does suggest that in a well-mixed system, like Fornax, there is no need for a separate DM component that does not follow the stellar profile. The second best candidate, which is also strongly disfavoured ($\Delta(\text{AICc}) \approx 12$), is a simple mass-follows-light model with a BH in the centre. We emphasize that these results should be verified with the use of proper Bayesian inference and the use of different tracer profiles; this is currently a work in progress.

We estimate a mass of $M_{\text{tot}} = 1.61 (^{+0.05}_{-0.035}) \times 10^8 M_{\odot}$ and a mass-to-light ratio $\Upsilon_V = 8.93 (^{+0.32}_{-0.20}) M_{\odot}/L_{\odot}$. These values are consistent with previous work (Walker et al. 2007; Lokas 2009; Jardel & Gebhardt 2012; Breddels & Helmi 2013); however, our mass-to-light ratio is more tightly constrained. Considering that we did not use the best photometric data available for the brightness profile and that we did not account for the non-negligible ellipticity $e \approx 0.3$ of Fornax, we expect that the value of the mass-to-light ratio and the total mass of the galaxy can be pushed to lower values.

We find an anisotropy profile that is tangentially biased. This tangential bias increases towards the outer parts of the dwarf; however our algorithm cannot tightly constrain the $\beta(r)$ anisotropy. This is because we fit directly for $\sigma_{\beta}^2$ and when we estimate $\beta(r)$ with the use of the Jeans equation, error propagation is significant. The tangential anisotropy seems to favour the scenario that Fornax has undergone a recent minor merger (Klimentowski et al. 2007; Yozin & Bekki 2012). Yozin & Bekki (2012) performed numerical simulations for Fornax and argued that an initially disk galaxy can be transformed through tidal stirring to a spheroid. Thus, this tangential anisotropy is perhaps a signature of an initial discy structure. Coleman et al. (2004) observed a shell structure that could favour this scenario; however, this is most likely an incorrect assumption, resulting from a misidentified overdensity of background galaxies (Bate et al. 2015).

7 FUTURE DIRECTION

The method we presented here has great potential for further applications. Some of our goals for our future work is to expand our modelling for triaxial systems. Other work in progress is to verify our method with outputs from numerical simulations (Diakogiannis et al., in preparation) and finally apply our algorithm to the classical dwarfs to perform mass modelling independent of the MAD.
ACKNOWLEDGEMENTS

We wish to thank our referee, Gary Mamon, for his many useful comments and suggestions that helped us improve the paper. Special thanks to Matthias Redlich for providing a cubic spline interpolation library, and Dan Taranu for careful reading of the manuscript. FID acknowledges support from ARC Discovery Project (DP140100198) and the Research Collaboration Awards (PG12105204). GFL acknowledges support from ARC Discovery Project (DP110100678) and Future Fellowship (FT100100268).

REFERENCES

Amorisco N. C., Agnello A., Evans N. W., 2013, MNRAS, 429, L89
An J. H., Evans N. W., 2006, ApJ, 642, 752
Bate M. F., McMonigal B., Lewis G. F., Irwin M. J., Gonzalez-Solares E., Shanks T., Metcalfe N., 2015, MNRAS, 453, 690
Battaglia G., Sollima A., Nipoti C., 2015, MNRAS, 454, 2401
Bicknell G. V., Bruce T. E. G., Carter D., Killeen N. E. B., 1989, ApJ, 336, 639
Binney J., 1980, MNRAS, 190, 873
Binney J., Mamon G. A., 1982, MNRAS, 200, 361
Binney J., Tremaine S., 2008, Galactic Dynamics: Second Edition. Princeton Univ. Press, Princeton, NJ
Braak C. J., 2006, Stat. Comput., 16, 239
Breddels M. A., Helmi A., 2013, A&A, 558, A35
Burkert A., 1995, ApJ, 447, L25
Burnham K. P., Anderson D. R., 2002, Model Selection and Multimodel Inference: A Practical Information-theoretic Approach. Springer, New York
Casella G., 1985, Am. Stat., 39, pp. 83
Cioffi L., Morganti L., 2010, MNRAS, 408, 1070
Coelho C. A. C., Lamont G. B., Veldhuizen D. A. V., 2006, Evolutionary Algorithms for Solving Multi-Objective Problems (Genetic and Evolutionary Computation). Springer-Verlag New York, Inc., Secaucus, NJ, USA
Coleman M., da Costa G. S., Martínez-Delgado D., Bland-Hawthorn J., 2004, in Prada F., Martínez del Gado D., Mahoney T. J., eds, ASP Conf. Ser. Vol. 327, Satellites and Tidal Streams, Astron. Soc. Pac., San Francisco, p. 173
Coleman M. G., Da Costa G. S., Bland-Hawthorn J., Freeman K. C., 2005, AJ, 129, 1443
Courteau S. et al., 2014, Rev. Modern Phys., 86, 47
de Blok W. J. G., McGaugh S. S., Rubin V. C., 2001, AJ, 122, 2396
Deb K., Anand A., Joshi D., 2002, Evol. Comput., 10, 371
den Hartog R., Katger P., 1996, MNRAS, 279, 349
Diakogiannis F. I., Lewis G. F., Ibata R. A., 2014a, MNRAS, 443, 598
Diakogiannis F. I., Lewis G. F., Ibata R. A., 2014b, MNRAS, 443, 610
Einasto J., Haud U., 1989, A&A, 223, 89
Evans N. W., An J., Walker M. G., 2009, MNRAS, 393, L50
Feroz F., Hobson M. P., Bridges M., 2009, MNRAS, 398, 1601
Ferreau H., Kirches C., Potschka A., Bock H., Diehl M., 2014, Math. Programm. Comput., 6, 327
Foreman-Mackey D. et al., 2016, corner.py: Scatterplot matrices in Python.
Available at: http://dx.doi.org/10.5281/zenodo.45906
Gay A., Dehuri S., Ghosh S., 2008, Multi-Objective Evolutionary Algorithms for Knowledge Discovery from Databases, Studies in Computational Intelligence. Springer Berlin Heidelberg
Gilmore G., Wilkinson M. I., Wyse R. F. G., Kleyna J. T., Koch A., Evans N. W., Grebel E. K., 2007, ApJ, 663, 948
Goldberg D. E., 1989, Genetic Algorithms in Search, Optimization and Machine Learning, 1st edn. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA
Goodman G. G., Wear J. J., 2010, CAMCoS, 5, 65
Hansen N., Ostermeier A., 2001, Completely Derandomized Self-Adaptation in Evolution Strategies. Vol. 9, MIT Press, Cambridge, MA, p. 159
Hansen N., Auger A., Ros R., Finck S., Poššk P., 2010, in Proceedings of the 12th Annual Conference Companion on Genetic and Evolutionary Computation, GECCO ’10. ACM, New York, NY, USA, p. 1689
Hastie T., Tibshirani R., Friedman J., 2001, Elements of Statistical Learning, Springer Series in Statistics. Springer New York Inc., New York, NY, USA
Hurvich C. M., Tsai C.-L., 1991, Biometrika, 78, 499
Ibata R., Nipoti C., Sollima A., Bellazzini M., Chapman S. C., Dalessandro E., 2013, MNRAS, 428, 3648
Irwin M., Hatzidimitriou D., 1995, MNRAS, 277, 1354
Jardel J. R., Gebhardt K., 2012, ApJ, 746, 89
King I. R., 1966, AJ, 71, 64
Klimentowski J., Łokas E. L., Kazantzidis S., Prada F., Mayer L., Mamon G. A., 2007, MNRAS, 378, 353
Łokas E. L., 2002, MNRAS, 333, 697
Łokas E. L., 2009, MNRAS, 394, L102
Mamon G. A., Boué G., 2010, MNRAS, 401, 2433
Mamon G. A., Łokas E. L., 2005, MNRAS, 362, 95
Mamon G. A., Biviano A., Boué G., 2013, MNRAS, 429, 3079
Mateo M. L., 1998, ARA&A, 36, 435
Merritt D., 1985, AJ, 90, 1027
Merritt D., 1987, ApJ, 313, 121
Merritt D., 2013, Dynamics and Evolution of Galactic Nuclei. Princeton Univ. Press, Princeton, p. 544
Merritt D., Graham A. W., Moore B., Diemand J., Terzić B., 2006, AJ, 132, 2685
Miklen R. A., Periaux J., Toivanen J., 1999, Int. J. Numer. Methods Fluids, 30, 149
Navarro J. F., Frenk C. S., White S. D. M., 1996, ApJ, 462, 563
Navarro J. F. et al., 2004, MNRAS, 349, 1039
Osipov P. L., 1979, Sov. Astron. Lett., 5, 42
Read J. I., Steger P., 2017, preprint (arXiv:prints)
Richardson T., Fairbairn M., 2013, MNRAS, 432, 3361
Rozenberg G., Bick T., Kok J. N., 2011, Handbook of Natural Computing, 1st edn. Springer Publishing Company, Inc.
Schwarzschild M., 1979, ApJ, 232, 236
Sugiyama N., 1978, Commun. Stat.-Theor. Methods, 7, 13
Talbi E.-G., 2009, Metaheuristics: From Design to Implementation. Wiley Publishing
Tiret O., Combès F., Angus G. W., Famaey B., Zhao H. S., 2007, A&A, 476, L1
Walcher C. J., Fried J. W., Burkert A., Klessen R. S., 2003, A&A, 406, 847
Walker M. G., Peñarrubia J., 2011, ApJ, 742, 20
Walker M. G., Mateo M., Olszewski E. W., Gnedin O. Y., Wang X., Sen B., Woodroffe M., 2007, ApJ, 667, L53
Walker M. G., Mateo M., Olszewski E. W., 2009a, AJ, 137, 3100
Walker M. G., Mateo M., Olszewski E. W., Sen B., Woodroffe M., 2009b, AJ, 137, 3109
Weinberg D. H., Bullock J. S., Governato F., Kuzio de Naray R., Peter A. H. G., 2015, Proc. Natl Acad. Sci., 112, 12249
Wolf J., 2010, Highlights Astron., 15, 79
Wolf J., 2011, in IAU Symposium, Vol. 271, IAU Symposium, Brunnell N. H., Brun A. S., Miesch M. S., Ponty Y., eds., p. 110
Yozin C., Bekki K., 2012, ApJ, 756, L18
Zhou A., Qu B.-Y., Li H., Zhao S.-Z., Suganthan P. N., Zhang Q., 2011, Swarm Evolutionary Comput., 1, 32

APPENDIX A: THE IMPORTANCE OF OPTIMUM KNOT DISTRIBUTION IN B-SPLINE FITTING

In this section, we give a simple example of a curve fitting to noiseless data in order to demonstrate the need for careful placement of knots. We create 1000 data points, uniformly distributed from a ‘difficult’ toy function

\[ f(x) = 6 + \cos(x^3)(2 + \exp\{\sin(x^2)\}) \]  

(A1)
We model this toy function by assuming the B-spline approximation of \( f(x) \approx \sum_{i=1}^{n} a_i B_i(x) \). The total number, \( n \), of unknown coefficients and the placement of the knots are evaluated with the use of the EA. For simplicity, we use no smoothing penalty and no errors. The EA evaluates the best fit with 24 knots, \( \xi \), appropriately placed with higher density in regions where the data demonstrate higher curvature. For comparison, we also fit with a uniform distribution, with 50 knots. We visualize our results in Fig. A1. Blue crosses correspond to the synthetic data. The purple line is the best EA recovered fit (24 knots). Green line is a fit with 50 uniform knots. Triangles pointing downwards correspond to the location of the knots, \( \xi \), of the EA evaluation, while triangles pointing upwards are the locations of the uniform knots. Clearly, the EA places knots in regions where the data demonstrate higher curvature.

APPENDIX B: COMPARISON OF MODELLING WITH UNIFORM KNOTS

In order to get a quantitative measure of the EA performance, we re-ran the EA for the case of Fornax for two models: the King const \( \Upsilon \) and the model with Burkert DM. We used a set of uniform knots, \( \xi = \{0, 0.25, 0.5, 0.75, 1\} \) scaled to \([0, 1]\). The King model again is chosen as the best model. The difference in AICc values becomes \( \Delta \text{AICc} = \text{AICc}_{\text{Burkert}} - \text{AICc}_{\text{King}} = 135.62 - 97.75 = 37.87 \), while with the EA-selected knots it is \( \Delta \text{AICc} = 27.85 \). Clearly, the differences in AICc between the knots chosen by the EA and the uniform ones are non-negligible (difference \( \approx 10 \)). We note that difference \( \geq 10 \) is conclusive for model selection when we use the AICc. This means that the EA helps to distinguish quantitatively between competing mass models. It also demonstrates that for comparison of models with fixed knots, the AICc difference can be conclusive while using the EA adaptive modelling, it may not.

Some more insight can be given by the values of the training ‘error’, \( \chi^2 = -\ln \mathcal{L} \) (equation 20). The King model gives \( \chi^2_{\text{EA}} = 39.0 \) with EA-selected knots, but with uniform knots it gives \( \chi^2_{\text{unif}} = 38.0 \), a lower value. This is expected since the uniform knots we used include also the EA-selected knots, but the basis is more flexible. That is, uniform knots tend to overfit the data. This becomes clear when we compare the AICc values of the EA-selected knots and the uniform ones: for the uniform knots, \( \text{AICc}_{\text{unif}} = 97.75 \), while for the EA-selected ones, \( \text{AICc}_{\text{EA}} = 86.50 \); therefore, the EA-selected model is better than the uniform knots one (\( \text{AICc}_{\text{EA}} < \text{AICc}_{\text{unif}} \)). The Burkert model gives \( \chi^2_{\text{EA}} = 38.36 \) with EA-selected knots, but \( \chi^2_{\text{unif}} = 38.82 \) with uniform knots. In this case, the uniform knots do not include the knot \( \xi \sim 0.122 \) that the EA finds; however, the basis is more flexible and the model has a tendency to overfit. Similar arguments hold for comparing the Burkert models (EA selected and uniform knots) with their AICc values.

Another quantitative measure of the EA performance can be seen by the fact that when we modelled Fornax, the EA gave solutions with the same number of knots – but different locations. These different models give significantly different AICc values, despite having the same number of unknown coefficients, \( a_i \). This is because the EA finds the best trade-off between the knots (numbers and location) and training error \( \chi^2 \) with the use of AICc. If we fix the number and location of the knots, as we did in Diakogiannis et al. (2014a,b), we lose the information of this trade-off.

APPENDIX C: EA TERMINOLOGY

This is a glossary of terms that is used in the EA literature for the aid of readers not familiar with EAs. It is by no means complete but it is focused in terminology we use in this work.

- **Individual** is a candidate solution, i.e. a set of variables. Sometimes the word chromosome is also used. In the simplest case, this is just a vector of real numbers, e.g. \((x_1, x_2, \ldots, x_n)\) this need not be the general case though. An individual can be any structure we can use as a representation inside a computer program that can be used to assign a score of quality (fitness value) to the individual.

- **Population** is a finite collection of individuals. For example, for the fitness function \( f(x, y) = x^2 + y^2 \), a population consists of a set of pair of values \( \{(x_i, y_i)\}, i = 1, \ldots, N_{\text{pop}} \), where \( N_{\text{pop}} \) is the total number of candidate solutions.

- **Generation**. In each iteration of the evolutionary algorithm, the software performs some operations in order to process the current population and produce a new one. The population at a specific iteration is called generation.

- **Fitness** of the individual is the value of a scalar function (fitness function) that can describe the quality of the solution. Individuals with highest fitness value describe a better fit (solution) to the problem.

- **Parents** are individuals that are chosen for creating new solutions from them.

- **Crossover** is the process of combining information from two or more individuals (parents) and produce one or more new individuals (offspring).

---

23 In the case of multiobjective optimization this can also be a vector function.
**Mutation.** The process of altering randomly an individual in order to produce a new solution.

**Recombination.** The process of crossover and mutation.

**Offspring** are individuals that are produced after the process of recombination.

**Selection** is the process of selecting individuals from the current generation that will be used as parents to produce new solutions (offspring).

**Evolutionary model** is the process that evolves the algorithm in each generation. This can be much more complicated than just selection and recombination. For example, we can have multiple offspring from one or more parents and we can assign an age to each individual who ceases to exist after a certain point, etc.

**APPENDIX D: CPU PROCESSING TIMES FOR FORNAX**

In the version of the algorithm used for this paper, we had approximately the following times:

(i) EA for a single model run: ~48 h.

(ii) For the optimum smoothing hyperparameters: ~24 h for a single model run.

(iii) For the MCMC ~48 h for the best model run.

The processing of Fornax took ~2 weeks. We need to emphasize though that the Phases 2 and 3 were applied only once, that is for the best selected model from Phase 1. The processing times may vary depending on how ‘difficult’ (how steep is the $\sigma^2_{\text{los}}$ profile) is a model to be fitted, as we found from experience with the GAIA CHALLENGE data set. These times are on a single laptop (HP ZBook).

This paper has been typeset from a \LaTeX file prepared by the author.