

SpheCow: Flexible dynamical models for galaxies and dark matter haloes

Maarten Baes[✉], Peter Camps[✉], and Bert Vandenbroucke[✉]

Sterrenkundig Observatorium, Universiteit Gent, Krijgslaan 281 S9, 9000 Gent, Belgium
e-mail: maarten.baes@ugent.be

Received 10 May 2021 / Accepted 25 May 2021

ABSTRACT

Simple but flexible dynamical models are useful for many purposes, including serving as the starting point for more complex models or numerical simulations of galaxies, clusters, or dark matter haloes. We present SpheCow, a new light-weight and flexible code that allows one to easily explore the structure and dynamics of any spherical model. Assuming an isotropic or Osipkov-Merritt anisotropic orbital structure, the code can automatically calculate the dynamical properties of any model with either an analytical density profile or an analytical surface density profile as starting point. We have extensively validated SpheCow using a combination of comparisons to analytical and high-precision numerical calculations, as well as the calculation of inverse formulae. SpheCow contains readily usable implementations for many standard models, including the Plummer, Hernquist, NFW, Einasto, Sérsic, and Nuker models. The code is publicly available as a set of C++ routines and as a Python module, and it is designed to be easily extendable, in the sense that new models can be added in a straightforward way. We demonstrate this by adding two new families of models in which either the density slope or the surface density slope is described by an algebraic sigmoid function. We advocate the use of the SpheCow code to investigate the full dynamical structure for models for which the distribution function cannot be expressed analytically and to explore a much wider range of models than is possible using analytical models alone.

Key words. methods: numerical – galaxies: structure – galaxies: kinematics and dynamics

1. Introduction

In order to investigate the structure and evolution of self-gravitating systems such as galaxies, clusters, and dark matter haloes, a characterisation of their structure in full 6D phase space is of paramount importance. Thanks to the improving observational data, novel modelling techniques, and increasing computational power, we now have various advanced dynamical modelling tools at our disposal (e.g., [Bovy 2015](#); [Vasiliev 2019](#); [Vasiliev & Valluri 2020](#); [Neureiter et al. 2021](#)). Nevertheless, relatively simple spherically symmetric models remain useful and important. They can act as an environment in which different physical processes can be investigated, or against which new modelling or data analysis techniques can be explored. Moreover, they are often the starting point for the creation of more complex and realistic models or full-scale numerical simulations.

Simple models have roughly become synonymous with analytical models, that is, models in which many properties can be calculated in terms of elementary or special functions. One approach to generate such simple models is by choosing a simple form for the distribution function. Once the distribution function is defined, the density profile $\rho(r)$ and potential $\Psi(r)$ of the self-consistent model can be determined by integrating the distribution function over velocity space and solving Poisson's equation. Famous models based on this approach include the singular isothermal sphere ([Binney & Tremaine 2008](#)), the King model ([King 1966](#)), and the Michie model ([Michie 1963](#); [Michie & Bodenheimer 1963](#)). Models with a simple analytical distribution function are very useful for elaborate calculations

in which the precise form of the distribution function is not the major uncertainty, for example to compute the rate of microlensing events in dark matter haloes ([Alcock et al. 1995](#)) or to predict the signal in direct dark matter detection experiments ([Evans et al. 2019](#)). The disadvantage of this approach is that the density and potential are often not analytically tractable and/or do not necessarily provide a realistic representation for actual galaxies or dark matter haloes.

As an alternative to this approach, it is also common to set up dynamical models with the mass density profile $\rho(r)$ as a starting point. If the density is sufficiently simple, other quantities can often be expressed analytically as well, such as the cumulative mass profile and the gravitational potential. For a number of popular models, such as the NFW model ([Navarro et al. 1997](#)) or the Einasto model ([Einasto 1965](#)), this is how far the analyticity goes ([Lokas & Mamon 2001](#); [Cardone et al. 2005](#)). For a fairly limited set of models, analytical expressions are also available for the phase-space distribution function, usually under the assumption of a simple orbital structure such as velocity isotropy. Some of the most popular spherical models belong to this category, such as the Plummer model ([Plummer 1911](#)), the isochrone sphere ([Hénon 1959](#)), the Jaffe model ([Jaffe 1983](#)), the Hernquist model ([Hernquist 1990](#)), and the family of γ -models ([Dehnen 1993](#); [Tremaine et al. 1994](#)). Finally, for a very restricted subset of these models more general anisotropic distribution functions and/or line profiles can be calculated analytically (e.g., [Dejonghe 1987](#); [Baes & Dejonghe 2002](#); [Baes & Van Hese 2007](#)).

A disadvantage of models with a simple density profile as starting point is that their surface density profile on the

plane of the sky¹ is not necessarily equally simple (e.g., [Zhao 1996](#); [Retana-Montenegro et al. 2012](#)). An interesting alternative is to consider models with a given surface density profile as a starting point. This approach has the advantage that specific functional forms can be chosen that closely reproduce the observed surface brightness distribution of galaxies and other systems. Examples of this type of models include the Sérsic model ([Sérsic 1968](#); [Graham & Driver 2005](#)) and the Nuker model ([Lauer et al. 1995](#)). The disadvantage is that the density needs to be calculated through an integral along the line of sight, and that this integration does not necessarily yield a simple expression. In particular, for the models listed above, the density profile can only be expressed in terms of very special functions ([Mazure & Capelato 2002](#); [Baes & Gentile 2011](#); [Baes & Van Hese 2011](#); [Baes 2020](#)). For other dynamical properties that typically require integrations of the density, finding explicit expressions is hopeless.

While completely analytical models have their obvious benefits, it would be a pity to only focus on the limited set of models that belong to this category. We advocate the use of other models as well, especially the much broader class of all models with an analytical density profile or surface density profile. Even when no analytical expressions are available for the most important dynamical properties, all of these properties can be calculated numerically in a relatively straightforward way.

In this paper we present a new light-weight and flexible code, SpheCow, that sets up a spherical dynamical model with either an analytical density profile $\rho(r)$ or an analytical surface density profile $\Sigma(R)$ as starting point. For either of these two options, the SpheCow code automatically calculates basic intrinsic and on-sky properties, as well as dynamical properties assuming either an isotropic or an Osipkov-Merritt type anisotropic orbital structure. This offers the possibility to easily calculate and investigate the full dynamical structure for models for which the distribution function cannot be expressed analytically, and to explore a much wider range of models than is possible using analytical models alone. In particular, such a light-weight tool can help to foster the use of more realistic models as starting points for more advanced modelling, at the expense of the simpler but often less realistic alternatives.

This paper is organised as follows. In Sect. 2 we discuss how the most important quantities, both intrinsic and on the plane of the sky, can be calculated when one has a closed expression for either the density (Sect. 2.1) or the surface density (Sect. 2.2). This section reviews many well-known equations, but also presents some shortcuts that can be used to convert double to single integrals. In Sect. 3 we present the SpheCow code in which we have implemented these formulae. We present the numerical integration algorithm that is the core of the code, and we describe the design of the code. In Sect. 4 we discuss the validation of the code. In Sect. 5 we present a simple application of the SpheCow code. We introduce two new general families of models, in which the logarithmic slope of either the mass density or the surface density profile is characterised by a sigmoid function. We discuss and summarise our results in Sect. 6.

2. The calculation of dynamical properties

We assume that a spherically symmetric model is either characterised by an analytical density profile $\rho(r)$ or an analytical

¹ Throughout this paper, we assume a constant mass-to-light ratio, such that surface brightness and surface density are equivalent quantities.

surface density profile $\Sigma(R)$. For either of these two cases, we discuss how the most important properties of the model can be calculated in an efficient way. In the remainder of this section, we assume that we have to our disposal a suitable numerical integration scheme that can efficiently calculate any integrals over semi-infinite or infinite intervals. This aspect will be discussed in Sect. 3.1 and is considered as a given for the time being.

2.1. Models defined by an analytical density profile

2.1.1. Basic properties

In most cases, the starting point of a model is the 3D mass density $\rho(r)$. If an analytical form of $\rho(r)$ is known, we can also directly analytically calculate the first-order and second-order derivatives, and therefore also the negative logarithmic density slope

$$\gamma(r) = -\frac{d \ln \rho}{d \ln r}(r) = -\frac{r \rho'(r)}{\rho(r)}. \quad (1)$$

With $\rho(r)$ given, we can calculate the total mass

$$M = 4\pi \int_0^\infty \rho(u) u^2 du, \quad (2)$$

the cumulative mass distribution

$$M(r) = 4\pi \int_0^r \rho(u) u^2 du, \quad (3)$$

and the positive gravitational potential

$$\Psi(r) = \frac{GM(r)}{r} + 4\pi G \int_r^\infty \rho(u) u du. \quad (4)$$

The circular velocity curve is

$$v_c(r) = \frac{GM(r)}{r}. \quad (5)$$

The total potential energy is calculated as ([Binney & Tremaine 2008](#); [Baes & Ciotti 2019a](#))

$$W_{\text{tot}} = -4\pi G \int_0^\infty \rho(u) M(u) u du. \quad (6)$$

The surface density profile can be calculated using an integration along the line of sight,

$$\Sigma(R) = 2 \int_R^\infty \frac{\rho(u) u du}{\sqrt{u^2 - R^2}}. \quad (7)$$

In a similar way as for the density profile, it is often interesting to characterise the negative logarithmic slope of the surface density profile (e.g., [Lauer et al. 1995, 2007](#)),

$$\gamma_p(R) = -\frac{d \ln \Sigma}{d \ln R}(R) = -\frac{R \Sigma'(R)}{\Sigma(R)}. \quad (8)$$

This expression requires the derivative of the surface density profile, for which we have, in general, no closed analytical expression. One way to automatically calculate it for an arbitrary model with a given density profile $\rho(r)$ is to directly apply numerical differentiation to the surface density profile calculated numerically through expression (7). A more convenient and accurate

way, however, is to directly take the formal derivative of expression (7). Using the substitution $u = R \sec \theta$ we have

$$\Sigma(R) = 2 \int_0^{\pi/2} \rho(R \sec \theta) R \sec^2 \theta d\theta. \quad (9)$$

Taking the derivative with respect to R and transforming it back to u , we find

$$\Sigma'(R) = 2 \int_R^\infty \frac{[\rho(u) + u \rho'(u)] u du}{R \sqrt{u^2 - R^2}}, \quad (10)$$

where $\rho'(r)$ is the first-order derivative of the density. The cumulative surface mass profile is the mass within a circular surface density contour on the sky with radius R ,

$$M_p(R) = 2\pi \int_0^R \Sigma(u) u du. \quad (11)$$

Inserting expression (7) and changing the order of the integration, we get (Mamon et al. 2010)

$$M_p(R) = M - 4\pi \int_R^\infty \rho(u) u \sqrt{u^2 - R^2} du. \quad (12)$$

This expression involves a combination of two single integrals, one of which only needs to be calculated once, rather than a double integration.

2.1.2. Dynamical properties for an isotropic orbital structure

It is well-known that each spherical potential-density pair corresponds to a unique dynamical model with an isotropic velocity distribution (Dejonghe 1986; Binney & Tremaine 2008). The velocity dispersion profile $\sigma_{\text{iso}}^2(r)$ of this model can be found through the solution of the Jeans equation,

$$\rho(r) \sigma_{\text{iso}}^2(r) = G \int_r^\infty \frac{\rho(u) M(u) du}{u^2}. \quad (13)$$

The total kinetic energy is calculated as

$$K_{\text{tot}} = 6\pi \int_0^\infty \rho(u) \sigma_{\text{iso}}^2(u) u^2 du. \quad (14)$$

The line-of-sight velocity or projected dispersion $\sigma_{\text{p,iso}}^2(R)$ is found as

$$\Sigma(R) \sigma_{\text{p,iso}}^2(R) = 2 \int_R^\infty \frac{\rho(u) \sigma_{\text{iso}}^2(u) u du}{\sqrt{u^2 - R^2}}. \quad (15)$$

We can introduce expression (13) into this equation and change the order of integration (Prugniel & Simien 1997). This yields the equivalent expression

$$\Sigma(R) \sigma_{\text{p,iso}}^2(R) = 2G \int_R^\infty \frac{\rho(u) M(u) \sqrt{u^2 - R^2} du}{u^2}. \quad (16)$$

The isotropic distribution function, which only depends on the (positive) binding energy per unit mass \mathcal{E} , can be found through Eddington's formula (Eddington 1916),

$$f_{\text{iso}}(\mathcal{E}) = \frac{1}{2\sqrt{2}\pi^2} \int_0^\mathcal{E} \frac{\tilde{\rho}''(\Psi) d\Psi}{\sqrt{\mathcal{E} - \Psi}}, \quad (17)$$

with $\tilde{\rho}(\Psi)$ the augmented density, that is the density written as a function of the potential. Following Binney (1982), we transform

this integral via the substitution $\Psi = \Psi(u)$, where we use the identity

$$\frac{d\Psi(r)}{dr} = -\frac{GM(r)}{r^2}. \quad (18)$$

This results in

$$f_{\text{iso}}(\mathcal{E}) = \frac{1}{2\sqrt{2}\pi^2} \int_{r_\mathcal{E}}^\infty \frac{\Delta(u) du}{\sqrt{\mathcal{E} - \Psi(u)}}, \quad (19a)$$

with

$$\Delta(r) = \frac{r^2}{GM(r)} \left[\rho''(r) + \rho'(r) \left(\frac{2}{r} - \frac{4\pi\rho(r)r^2}{M(r)} \right) \right], \quad (19b)$$

and $r_\mathcal{E}$ implicitly defined by the equation $\Psi(r_\mathcal{E}) = \mathcal{E}$.

The differential energy distribution represents the distribution of mass as a function of the binding energy \mathcal{E} . For isotropic systems, the differential energy distribution can be written as

$$N_{\text{iso}}(\mathcal{E}) = f_{\text{iso}}(\mathcal{E}) g_{\text{iso}}(\mathcal{E}), \quad (20)$$

with $g_{\text{iso}}(\mathcal{E})$ the density-of-states function, defined as

$$g_{\text{iso}}(\mathcal{E}) = -16\sqrt{2}\pi^2 \int_\mathcal{E}^{\Psi_0} r^2 \frac{dr}{d\Psi} \sqrt{\Psi - \mathcal{E}} d\Psi \quad (21)$$

with $r = r(\Psi)$ the radius written as a function of the binding potential, and Ψ_0 the depth of the potential well. This equation can be recast in the form

$$g_{\text{iso}}(\mathcal{E}) = 16\sqrt{2}\pi^2 \int_0^{r_\mathcal{E}} u^2 \sqrt{\Psi(u) - \mathcal{E}} du. \quad (22)$$

2.1.3. Dynamical properties for an Osipkov-Merritt orbital structure

An isotropic orbital structure is the simplest option and it is interesting from a mathematical point of view. However, galaxies and dark matter haloes are not necessarily characterised by this simple orbital structure. In fact, most numerical N -body simulations predict that dark matter haloes are roughly isotropic in the central regions, but radially anisotropic in the outer regions (Taylor & Navarro 2001; Diemand et al. 2004; Ludlow et al. 2011; Lemze et al. 2012; Wojtak et al. 2013; Butsyk et al. 2016).

Osipkov (1979) and Merritt (1985a) independently discovered that dynamical systems with a spheroidal velocity distribution have the interesting property that their distribution function can be calculated in a similar way as models with an isotropic distribution function. The now so-called Osipkov-Merritt models are characterised by an orbital structure that is isotropic in the centre and completely radially anisotropic at large radii. More concretely, the anisotropy profile $\beta(r)$ is

$$\beta(r) \equiv 1 - \frac{\sigma_\theta^2(r)}{\sigma_r^2(r)} = \frac{r^2}{r^2 + r_a^2}, \quad (23)$$

with r_a a freely chosen anisotropy radius. In this case, the velocity dispersions are obviously not the same in the three principle directions. For a given density profile and anisotropy radius, the radial velocity dispersion profile can be calculated as (Mamon & Łokas 2005)

$$\rho(r) \sigma_{\text{r,om}}^2(r) = \frac{Gr_a^2}{r^2 + r_a^2} \int_r^\infty \frac{\rho_Q(u) M(u) du}{u^2}, \quad (24)$$

with $\rho_Q(r)$ defined as

$$\rho_Q(r) = \left(1 + \frac{r^2}{r_a^2}\right) \rho(r). \quad (25)$$

The velocity dispersions in the two tangential directions are given by

$$\sigma_{\theta, \text{om}}^2(r) = \sigma_{\phi, \text{om}}^2(r) = \frac{r_a^2}{r^2 + r_a^2} \sigma_{r, \text{om}}^2(r). \quad (26)$$

In this case, the total kinetic energy is calculated as

$$\begin{aligned} K_{\text{tot}} &= 2\pi \int_0^\infty \rho(u) \left[\sigma_{r, \text{om}}^2(u) + \sigma_{\theta, \text{om}}^2(u) + \sigma_{\phi, \text{om}}^2(u) \right] u^2 du \\ &= 2\pi \int_0^\infty \left(\frac{u^2 + 3r_a^2}{u^2 + r_a^2} \right) \rho(u) \sigma_{r, \text{om}}^2(u) u^2 du. \end{aligned} \quad (27)$$

The general expression for the projected velocity dispersion for an anisotropic spherical model is (Binney & Mamon 1982; Dejonghe 1987)

$$\Sigma(R) \sigma_p^2(R) = 2 \int_R^\infty \left[1 - \frac{R^2}{u^2} \beta(u) \right] \frac{\rho(u) \sigma_r^2(u) u du}{\sqrt{u^2 - R^2}}, \quad (28)$$

which in the case of an Osipkov-Merritt orbital structure reduces to

$$\Sigma(R) \sigma_{p, \text{om}}^2(R) = 2 \int_R^\infty \left(\frac{u^2 + r_a^2 - R^2}{u^2 + r_a^2} \right) \frac{\rho(u) \sigma_{r, \text{om}}^2(u) u du}{\sqrt{u^2 - R^2}}. \quad (29)$$

In principle, this expression can directly be implemented, but the integrand depends on $\sigma_{r, \text{om}}^2(u)$, implying that we get at least a double integral (and a triple integral if the mass profile needs to be calculated numerically). A more convenient form can be obtained by substituting Eq. (24) and changing the order of integration in the resulting expression (Mamon & Łokas 2005; Agnello et al. 2014). This results in

$$\Sigma(R) \sigma_{p, \text{om}}^2(R) = G \int_R^\infty \frac{w(u, R) \rho(u) M(u) du}{u^2}, \quad (30a)$$

with

$$\begin{aligned} w(u, R) &= \left(\frac{u^2 + r_a^2}{R^2 + r_a^2} \right) \\ &\times \left(\frac{R^2 + 2r_a^2}{\sqrt{R^2 + r_a^2}} \arctan \sqrt{\frac{u^2 - R^2}{R^2 + r_a^2}} - \frac{R^2 \sqrt{u^2 - R^2}}{u^2 + r_a^2} \right). \end{aligned} \quad (30b)$$

The distribution function of anisotropic dynamical models can generally written as $f(\mathcal{E}, L)$ with L the total angular momentum per unit mass. For the special case of an Osipkov-Merritt orbital structure, the distribution function depends on \mathcal{E} and L only through the combination

$$Q = \mathcal{E} - \frac{L^2}{2r_a^2}, \quad (31)$$

that is, $f_{\text{om}}(\mathcal{E}, L) = f_{\text{om}}(Q)$, with the additional constraint that $f_{\text{om}}(Q) = 0$ for $Q \leq 0$. For a given density profile, the function $f_{\text{om}}(Q)$ can be determined using a formula very similar to the Eddington relation (Osipkov 1979; Merritt 1985a),

$$f_{\text{om}}(Q) = \frac{1}{2\sqrt{2}\pi^2} \int_0^Q \frac{\tilde{\rho}'_Q(\Psi) d\Psi}{\sqrt{Q - \Psi}}. \quad (32)$$

As for the isotropic case, this expression can be recast in a form that is more suitable for numerical integration,

$$f_{\text{om}}(Q) = \frac{1}{2\sqrt{2}\pi^2} \int_{r_Q}^\infty \frac{\Delta_Q(u) du}{\sqrt{Q - \Psi(u)}}, \quad (33a)$$

with r_Q determined implicitly by $\Psi(r_Q) = Q$, and

$$\Delta_Q(r) = \frac{r^2}{GM(r)} \left[\rho''_Q(r) + \rho'_Q(r) \left(\frac{2}{r} - \frac{4\pi\rho_Q(r)r^2}{M(r)} \right) \right]. \quad (33b)$$

A general calculation of the differential energy distribution for Osipkov-Merritt orbital structure is non-trivial. However, one can define a probability density function $\mathcal{N}_{\text{om}}(Q)$ for the mass as a function of Q , which we refer to as the pseudo-differential energy distribution. Cuddeford (1991) demonstrated that it can be found as

$$\mathcal{N}_{\text{om}}(Q) = f_{\text{om}}(Q) g_{\text{om}}(Q), \quad (34)$$

with $g_{\text{om}}(Q)$ a pseudo-density-of-states function,

$$g_{\text{om}}(Q) = 16\sqrt{2}\pi^2 \int_0^{r_Q} u^2 \left(1 + \frac{u^2}{r_a^2} \right)^{-1} \sqrt{\Psi(u) - Q} du. \quad (35)$$

2.2. Models defined by an analytical surface density profile

As discussed in the Introduction, another class of models starts from an observed surface density profile $\Sigma(R)$, with R the radius on the plane of the sky. As can be seen in the previous section, the fundamental quantity that appeared in nearly all the formulae is the density $\rho(r)$. The connection between density and surface density is given by Eq. (7), which can be inverted using the standard Abel inversion formula (Binney & Tremaine 2008; Mamon & Boué 2010)

$$\rho(r) = -\frac{1}{\pi} \int_r^\infty \frac{\Sigma'(u) du}{\sqrt{u^2 - r^2}}. \quad (36)$$

In this formula, $\Sigma'(R)$ is the derivative of the surface density profile, which is also known analytically since we assumed that $\Sigma(R)$ is an analytical function. For a given analytical surface density profile, we can numerically integrate expression (36) to obtain $\rho(r)$, and subsequently use this expression in all other expression. However, we can avoid the use of numerical double integrals for the calculation of the mass profile and gravitational potential by inserting the expression (36) into the general Eqs. (3) and (4), and changing the order of integrations. After some manipulation we find for the mass profile

$$M(r) = -\pi \left[\int_0^r \Sigma'(u) u^2 du + \int_r^\infty \Sigma'(u) w_-(u, r) du \right], \quad (37a)$$

where we have set

$$w_-(u, r) = \frac{2}{\pi} \left[u^2 \arctan \left(\frac{r}{\sqrt{u^2 - r^2}} \right) - r \sqrt{u^2 - r^2} \right]. \quad (37b)$$

For the gravitational potential we find in a similar way

$$\Psi(r) = -\frac{\pi G}{r} \left[\int_0^r \Sigma'(u) u^2 du + \int_r^\infty \Sigma'(u) w_+(u, r) du \right], \quad (38a)$$

with now

$$w_+(u, r) = \frac{2}{\pi} \left[u^2 \arctan \left(\frac{r}{\sqrt{u^2 - r^2}} \right) + r \sqrt{u^2 - r^2} \right]. \quad (38b)$$

For all other properties, corresponding to either an isotropic or an Osipkov-Merritt orbital distribution, we directly use the expressions from the previous subsection, in which we substitute the (numerically determined) expressions (36)–(38b) for density, mass profile, and potential. The only additional quantities that we need are the first and second order derivatives of the density, which are necessary to calculate the slope of the density profile and the distribution function. Rather than applying numerical differentiation to the numerically determined density profile (36), it is more straightforward and far more accurate to use the expressions

$$\rho'(r) = -\frac{1}{\pi} \int_r^\infty \frac{\Sigma''(u) u \, du}{r \sqrt{u^2 - r^2}}, \quad (39)$$

$$\rho''(r) = -\frac{1}{\pi} \int_r^\infty \frac{\Sigma'''(u) u^2 \, du}{r^2 \sqrt{u^2 - r^2}}, \quad (40)$$

with $\Sigma''(R)$ and $\Sigma'''(R)$ the second and third order derivatives of the surface density profile, for which we assume exact expressions are available. These formulae can be obtained from Eq. (36) in a similar way as we derived Eq. (10) from Eq. (7).

3. The SpheCow code

We have designed a C++ software module called SpheCow to automatically calculate the dynamical properties as described above for any spherical model. Models can be defined either by their density profile, or by their surface density profile. In both cases, the code can be used to calculate the basic model properties and the most important dynamical properties, for both isotropic and Osipkov-Merritt-type anisotropic orbital structures.

3.1. Numerical integrations

At the core of the SpheCow code is a generic numerical integration scheme that is appropriate for all models, whether they are based on density or surface density. As can be seen from the set of formulae in the previous section, we essentially encounter only two types of integrals: all integrations run over the variable u , which can represent either the spherical radius or the projected radius on the plane of the sky, and they either run on an interval $[0, r]$ or an interval $[r, \infty]$. So in principle we only have to implement a strategy to numerically calculate such integrals. A more refined strategy is required, however, to ensure that we get accurate results for a wide variety of galaxy models. Realistic models are typically characterised by a particular scale radius r_b , which often corresponds to the division between an inner and an outer profile. It therefore seems logical to split any integration over an interval that contains r_b into two separate integrals. This is particularly necessary for models in which there is a (relatively) sharp break in the density profile at $r = r_b$.

This implies that we need to consider four different types of integrals, namely integrals on the intervals $[0, r]$ or $[r, r_b]$ if $r < r_b$, or on the intervals $[r_b, r]$ or $[r, \infty]$ if $r > r_b$. For each of these integrals we use a substitution to turn it into an equivalent integration that is more suitable for a Gauss-Legendre quadrature scheme. In the case $r < r_b$, we use

$$\int_0^r X(u) \, du = r \int_0^{\pi/2} X(r \sin \theta) \cos \theta \, d\theta, \quad (41)$$

$$\int_r^{r_b} X(u) \, du = r \int_{\arcsin(r/r_b)}^{\pi/2} X(r \csc \theta) \cos \theta \csc^2 \theta \, d\theta. \quad (42)$$

For $r > r_b$ we use similar transformations,

$$\int_{r_b}^r X(u) \, du = r \int_{\arcsin(r_b/r)}^{\pi/2} X(r \sin \theta) \cos \theta \, d\theta, \quad (43)$$

$$\int_r^\infty X(u) \, du = r \int_0^{\pi/2} X(r \csc \theta) \cos \theta \csc^2 \theta \, d\theta. \quad (44)$$

For integrals covering the entire intervals $[0, r_b]$ and $[r_b, \infty]$, we use the transformations (41) and (44), respectively, with $r \rightarrow r_b$,

$$\int_0^{r_b} X(u) \, du = r_b \int_0^{\pi/2} X(r_b \sin \theta) \cos \theta \, d\theta, \quad (45)$$

$$\int_{r_b}^\infty X(u) \, du = r_b \int_0^{\pi/2} X(r_b \csc \theta) \cos \theta \csc^2 \theta \, d\theta. \quad (46)$$

To perform the actual integrations, we use a Gauss-Legendre quadrature formula. The integration points and the corresponding weights are pre-calculated, which makes the integrations fast and efficient. We use 128 integration points as our default, but this value can be chosen by the user. In general, higher orders correspond to higher accuracy but to an increased calculation time as well (see Sect. 4.1 for a discussion).

3.2. Code design

For the design of the SpheCow code we have deliberately chosen a modular and light-weight setup. The code essentially consists of a simple driver function, a GaussLegendre class that takes care of the numerical integrations, and a general abstract Model class that acts as the base class for all different models. This last class has two abstract subclasses, DensityModel and SurfaceDensityModel. The former is the base class for all different models defined by an analytical density profile, the latter for all models with an analytical surface density profile as starting point.

Once the density or surface density and their derivatives are known, the calculation of all the photometric and dynamical properties is calculated via a direct implementation of the formulae as written in the previous section. The only aspect where some special care is needed is the integrations that contain a factor $\Psi(r) - \Psi(u)$ in the denominator in the integrand. When $u - r = \epsilon$ is very small, this difference can effectively become zero and this can cause infinities. In this case we use the first two terms in the Taylor expansion,

$$\Psi(r) - \Psi(r + \epsilon) \approx \frac{GM(r)}{r^2} \epsilon + \left[2\pi \rho(r) - \frac{GM(r)}{r^3} \right] \epsilon^2. \quad (47)$$

We have currently implemented a suite of models, listed in Table 1. There are many other models in the literature that we have not (yet) included in the code, including the generalised isochrone sphere (An & Evans 2006a), the core-NFW model (Peñarrubia et al. 2012), the doubloon models (Evans & Williams 2014; Evans et al. 2015), or the generalised Einasto model (Fielder et al. 2020). However, the SpheCow code is modular in design, and new models can easily be added, as separate subclasses that inherit from either the DensityModel and SurfaceDensityModel base classes. To add a model with an analytical density profile, the user needs to provide an implementation of the density, and its first and second order derivatives. All other quantities can be calculated numerically using the formulae and numerical integration schemes discussed above. However, in case closed analytical expressions are available, in particular for the mass profile or the potential, these can also

Table 1. Models currently implemented in the SpheCow code.

Name	SpheCow class	References
Models defined by $\rho(r)$		
	DensityModel	
Broken power-law	BPLModel	Du et al. (2020), Baes & Camps (2021)
Burkert	BurkertModel	Burkert (1995), Salucci & Burkert (2000)
Einasto	EinastoModel	Einasto (1965), Cardone et al. (2005)
γ -model	GammaModel	Dehnen (1993), Tremaine et al. (1994), Carollo et al. (1995), Baes & Dejonghe (2004)
Hernquist	HernquistModel	Hernquist (1990), Baes & Dejonghe (2002)
Hypervirial	HypervirialModel	Veltmann (1979), Evans & An (2005)
Isochrone	IsochroneModel	Hénon (1959, 1960), May & Binney (1986)
Jaffe	JaffeModel	Jaffe (1983), Merritt (1985b)
Navarro, Frenk & White	NFWModel	Navarro et al. (1997), Łokas & Mamon (2001)
Perfect sphere	PerfectSphereModel	de Zeeuw (1985)
Plummer	PlummerModel	Plummer (1911), Dejonghe (1987), Merritt (1985a), Cuddeford (1991)
Zhao	ZhaoModel	Zhao (1996)
Models defined by $\Sigma(R)$		
	SurfaceDensityModel	
de Vaucouleurs	DeVaucouleursModel	de Vaucouleurs (1948), Binney (1982)
Nuker	NukerModel	Lauer et al. (1995), Baes (2020)
Sérsic	SersicModel	Sérsic (1968), Graham & Driver (2005), Ciotti (1991), Baes & Ciotti (2019b)

Notes. Models defined by an analytical density profile are subclasses of the general `DensityModel` class, models defined by an analytical surface density profile are subclasses of the general `SurfaceDensityModel` class.

be provided. This will help the calculation of other properties such as the distribution function, both in speed and accuracy. Similarly, to add a new model with an analytical surface density profile, a new subclass of the `SurfaceDensityModel` class must be generated, and an implementation of the surface density profile $\Sigma(R)$, and its first, second, and third order derivatives are required. All other quantities can be calculated automatically.

3.3. Availability

SpheCow is publicly available on GitHub² as a set of C++ files. A Makefile and a readme file are available to help users with the installation and running of the code. We have also foreseen Python bindings and a Python script to convert the SpheCow code to a Python module that can directly be used in Python code and notebooks. An Jupyter Notebook example is included to help users to get started.

4. Validation

4.1. Accuracy of the numerical integrations

We have extensively validated the SpheCow code. A first important aspect is a characterisation of the accuracy of the calculations, which we have done by comparing the results of the code against analytical results for a number of quantities and models where such analytical expressions exist. Very useful models for which analytical expressions exist for nearly all relevant quantities, both for an isotropic and an Osipkov-Merritt orbital structure, are the Plummer model, the Hernquist model, and the γ -model with $\gamma = \frac{5}{2}$. For most other models implemented, at least some properties can be calculated analytically: analytical formulae can be found in the articles referenced in the last

² <https://github.com/mbaes/SpheCow>

column of Table 1. We have compared the numerical results against the analytical ones where possible, and generally found excellent agreement.

In Fig. 1 we illustrate the accuracy of the SpheCow code by means of the Plummer model. In the nine panels we show the absolute value of the relative errors between the SpheCow calculations and extended precision calculations based on the analytical formulae from Merritt (1985a) and Dejonghe (1987). The top row shows three basic properties: the density, the surface density, and the potential. The middle row shows the intrinsic dispersion, the projected dispersion, and the distribution function corresponding to an isotropic orbital structure, whereas the bottom row shows the same properties, but now for an Osipkov-Merritt orbital structure with $r_a = c$, with c the scale radius of the Plummer model.

The thick yellow curves correspond to the Plummer model as implemented in the SpheCow code. It is a subclass of the `DensityModel` class, that is, it is defined by means of its analytical density profile. Since the Plummer model also has simple analytical expressions for the mass profile and the gravitational potential, these are also implemented in the `PlummerModel` class, and this explains why the density and the potential have relative errors of the order of machine precision. The surface density profile, the intrinsic and projected dispersion profiles, and the distribution functions all involve just a single numerical integration. For the surface density and projected dispersion, the relative error is of the order of 10^{-12} ; for the intrinsic dispersion and the distribution functions, the relative error is even smaller.

The thin black curves in Fig. 1 also represent SpheCow calculations for a Plummer model, but now we have not used the specific Plummer implementations for the potential and mass profile. Both are now calculated numerically using our standard Gauss-Legendre integration scheme based on Eqs. (3) and (4), and this results in a relative error of the order of 10^{-13} . This

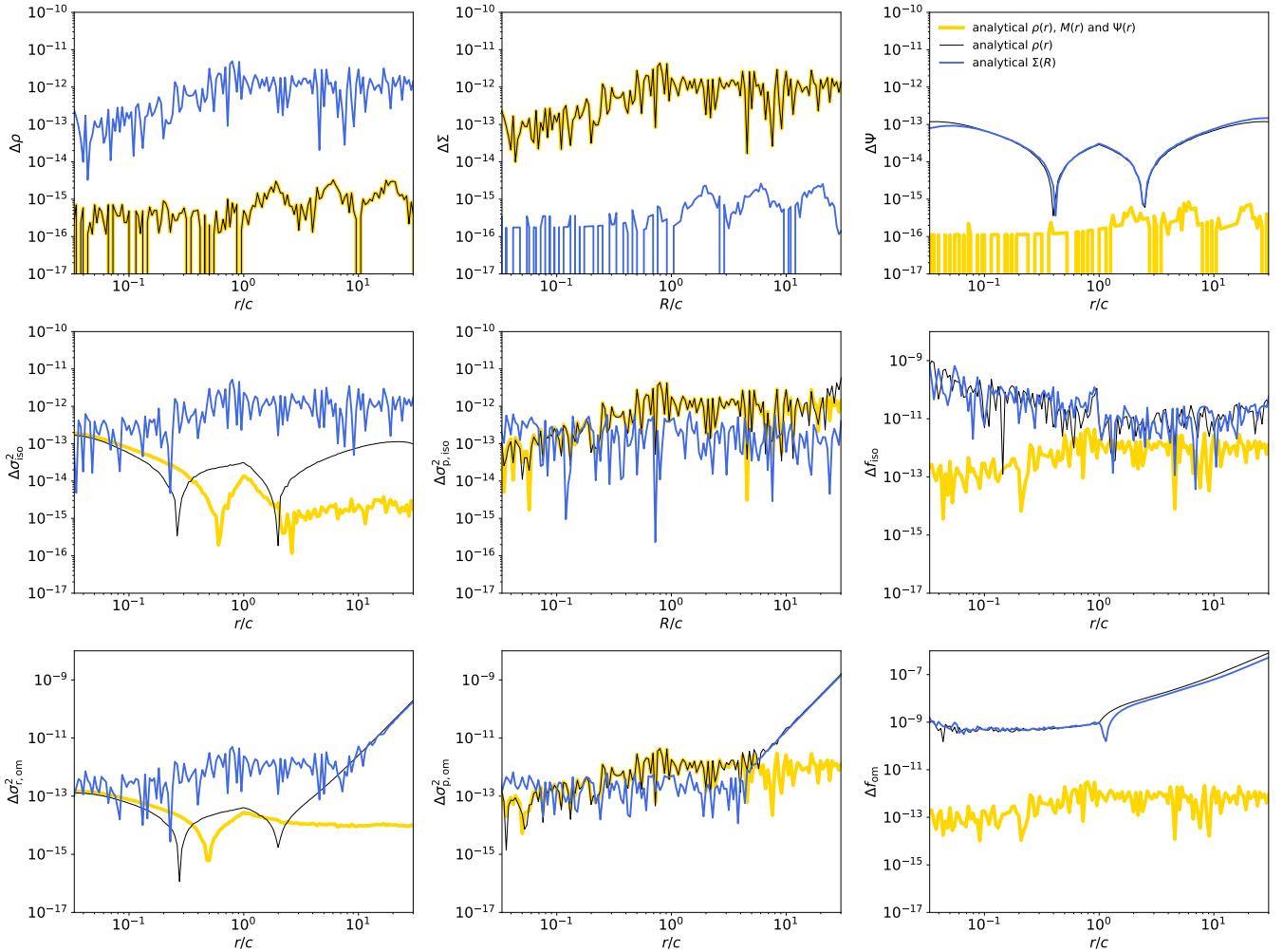


Fig. 1. Absolute value of the relative error of the density, surface density, potential, velocity dispersion, projected dispersion, and distribution function of the Plummer model (c represents the scale radius of the Plummer model). See text in Sect. 4.1 for the meaning of the different curves in each panel.

does not affect the surface density profile and it only marginally increases the relative error of the dispersion profiles for the isotropic model. It does affect the accuracy of the dispersion profiles of the models with an Osipkov-Merritt orbital structure, especially at the outer, radially anisotropic regions. It also affects the relative error of the isotropic and Osipkov-Merritt distribution functions. For the isotropic case, the distribution function is now characterised by a relative error of the order of 10^{-10} , especially at small radii. For the anisotropic distribution function, we see a larger loss of accuracy. In the inner, isotropic regions we have a relative error of about 10^{-9} , in the outer radially anisotropic region, the relative error increases up to 10^{-6} .

Finally, we have also implemented an alternative version of the Plummer model in SpheCow, now as subclass of the `SurfaceDensityModel` class. Indeed, the Plummer model is quite unique in that it also has a very simple analytical surface density profile. The results of this alternative model are shown as the blue lines in Fig. 1. In this case, the density profile, mass profile, and potential all involve a single numerical integration, as shown in Eqs. (36)–(38b). For the density, this results in a relative error of the order of 10^{-12} , for the potential the accuracy is an order of magnitude better. Both the intrinsic and the line-of-sight dispersion profiles now require a double numerical integration, but still the relative error remains at the order

of 10^{-12} for the intrinsic dispersion and even around 10^{-13} for the projected dispersion (except for the Osipkov-Merritt case at large radii). The relative error on the distribution functions does not alter significantly.

For Fig. 1 we used our default Gauss-Legendre integration scheme with 128 nodes, but this number can be chosen freely by the user, as discussed in Sect. 3.1. In Fig. 2 we show how the relative error on the surface density, potential, and distribution function changes as a function of the number of integration points in the numerical integration scheme. To get a single representative relative error, we calculated the relative error corresponding to 201 points spread logarithmically between $r/r_b = 10^{-2}$ and 10^2 , and we took the average value of the absolute values. We used a Plummer model without an explicit implementation of the potential and the mass profile for the calculations, that is the model corresponding to the thin black lines in Fig. 1. The number of integration points is varied from 8 to 512.

For the surface density and the potential, quantities that require just one and two individual integrations respectively, the accuracy of the SpheCow calculation increases spectacularly as the number of integration points increases. For example, increasing this number from 32 to 64, the relative error decreases by roughly five orders of magnitude. As soon as the number of points reaches about 128, the accuracy does not improve any

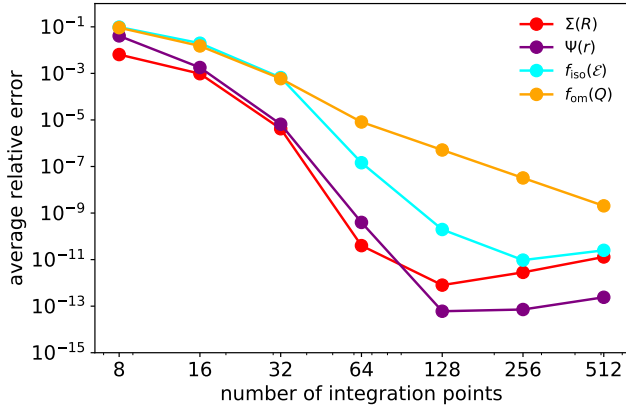


Fig. 2. Average relative error on the calculation of the surface density, potential, and distribution function for the Plummer model as a function of the number of integration points in the Gauss-Legendre integration scheme.

further. The calculation of the distribution function, both for an isotropic and an Osipkov-Merritt orbital structure, is more complex and involves nested numerical integrations. The accuracy of the calculations still increases strongly with increasing number of integration points, but not as outspoken as for the surface density and potential. The accuracy also keeps increasing beyond 128 integration points.

In general, Fig. 2 and additional tests suggest that a Gauss-Legendre integration scheme with 128 integration points represents a good compromise between high accuracy and reasonable computation time (for quantities that involve a double integral, such as the distribution function, the computation time scales quadratically with the number of integration points). In case a reduced accuracy is sufficient, a slightly lower order can be considered.

4.2. Consistency checks

The SpheCow code already contains a significant number of models and is designed to be extended with additional models in a straightforward way. Each new model requires the implementation of either the density and its first two derivatives, or the surface density and its first three derivatives. In order to check the consistency of the implementation of these properties, we can use the inverse formulae for many of the quantities that are being calculated. For example, the distribution function represents the density of the model in 6D phase space, and the mass density profile can be recovered by integrating the distribution function over velocity space. In the case of an isotropic orbital structure, we find

$$\rho(r) = 4\sqrt{2}\pi \int_0^{\Psi(r)} f_{\text{iso}}(\mathcal{E}) \sqrt{\Psi(r) - \mathcal{E}} d\mathcal{E}, \quad (48)$$

and for an Osipkov-Merritt orbital structure we have (Merritt 1985a)

$$\rho(r) = 4\sqrt{2}\pi \left(1 + \frac{r^2}{r_a^2}\right)^{-1} \int_0^{\Psi(r)} f_{\text{om}}(Q) \sqrt{\Psi(r) - Q} dQ. \quad (49)$$

In a similar way, the velocity dispersion profiles can be found as the second-order moment of the distribution function, resulting in

$$\rho(r) \sigma_{\text{iso}}^2(r) = \frac{8\sqrt{2}\pi}{3} \int_0^{\Psi(r)} f_{\text{iso}}(\mathcal{E}) [\Psi(r) - \mathcal{E}]^{3/2} d\mathcal{E}, \quad (50)$$

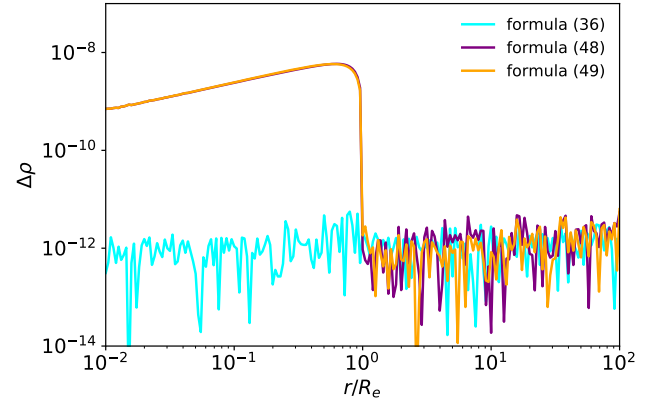


Fig. 3. Absolute value of the relative error of the density of the de Vaucouleurs model, as calculated by SpheCow. The cyan line corresponds to the calculation using the standard deprojection Eq. (36), the purple line to the integration of the isotropic distribution function over velocity space (48), and the orange line to the integration of the Osipkov-Merritt distribution function over velocity space (49).

and

$$\rho(r) \sigma_{\text{r,om}}^2(r) = \frac{8\sqrt{2}\pi}{3} \left(1 + \frac{r^2}{r_a^2}\right)^{-1} \times \int_0^{\Psi(r)} f_{\text{om}}(Q) [\Psi(r) - Q]^{3/2} dQ. \quad (51)$$

The four Eqs. (48)–(51) can easily be converted to integrations with respect to radius. For example, Eq. (48) can be transformed into

$$\rho(r) = 4\sqrt{2}\pi G \int_r^\infty \frac{f_{\text{iso}}(\Psi(u)) M(u) \sqrt{\Psi(r) - \Psi(u)} du}{u^2}. \quad (52)$$

We have implemented all of these inverse formulae in the SpheCow code to be used as a way to validate the implementation of the different models.

Moreover, they can also be used as a way to characterise the accuracy of the numerical integrations. In Fig. 3 we show the absolute value of the relative error for the density of the de Vaucouleurs model. This model is implemented in the SpheCow by means of its analytical surface density profile, and the cyan curves correspond to the density calculated via the standard deprojection formula (36). Comparing these values to the exact values, which can only be expressed exactly in terms of non-standard special functions (Mazure & Capelato 2002; Baes & Gentile 2011), we see that the relative errors are of the order of 10^{-12} , similar to what we obtained for the Plummer model in Fig. 1. The purple lines in Fig. 3 correspond to the density as calculated by integrating the isotropic distribution function over velocity space, that is using formula (48). The orange lines are similar, but now the Osipkov-Merritt distribution function is integrated over velocity according to Eq. (49), with $r_a/R_e = 2$. For both the isotropic and the Osipkov-Merritt cases, the relative error on the density for $r > R_e$ is of the same magnitude as the relative error of the density as calculated through the standard deprojection formula. This is remarkable, given that the integrands in the formulae (48) and (49) are a function of the distribution function, the mass profile and the potential, each of which are obtained via a numerical integration itself (the distribution function even via a complicated double integration). For $r < R_e$, we see that there is a systematic increase of

the relative error up to about 10^{-9} , because the integration now involves a combination of two separate integrals.

Finally, we can also test the implementation of the different models through the differential energy distribution. For a model with an isotropic orbital structure, $\mathcal{N}(\mathcal{E})$ represents the distribution of mass as a function of the binding energy, and it should therefore satisfy the normalisation

$$\int_0^{\Psi_0} \mathcal{N}_{\text{iso}}(\mathcal{E}) d\mathcal{E} = G \int_0^{\infty} \frac{f_{\text{iso}}(\Psi(u)) g_{\text{iso}}(\Psi(u)) M(u) du}{u^2} = M. \quad (53)$$

Similarly, we have

$$\int_0^{\Psi_0} \mathcal{N}_{\text{om}}(Q) dQ = G \int_0^{\infty} \frac{f_{\text{om}}(\Psi(r)) g_{\text{om}}(\Psi(u)) M(u) du}{u^2} = M. \quad (54)$$

All models currently in the SpheCow code have successfully passed these tests (for different parameters for those models that depend on different parameters). We strongly recommend any user who adds a new model to the SpheCow code, something we strongly encourage, to perform these basic tests as well.

5. Sigmoid models

5.1. Density sigmoid models

As an application and a demonstration of the power of the SpheCow code, we develop a number of new families of spherical models. Many models, both for galaxies and dark matter haloes, are characterised by a density profile with a power-law behaviour at small and large radii, that is

$$\rho(r) \propto r^{-\gamma} \quad \text{for } r \ll r_b, \quad (55a)$$

$$\rho(r) \propto r^{-\beta} \quad \text{for } r \gg r_b. \quad (55b)$$

with $r = r_b$ some break radius. An equivalent condition is that the logarithmic density slope $\gamma(r)$ behaves as

$$\gamma(r) \approx \gamma \quad \text{for } r \ll r_b, \quad (56a)$$

$$\gamma(r) \approx \beta \quad \text{for } r \gg r_b. \quad (56b)$$

Furthermore, it is desirable that the logarithmic density slope changes smoothly and monotonically between the asymptotic values γ and β . Almost all of the models in the first part of the list in Table 1 satisfy these criteria. A general and flexible model that was proposed with exactly these criteria is the Zhao or double power-law model, presented by Zhao (1996). This model is used extensively to model dark matter haloes (e.g., Hague & Wilkinson 2013, 2014, 2015; Di Cintio et al. 2014; Allaert et al. 2017; Dekel et al. 2017; Freundlich et al. 2020a,b; Hayashi et al. 2020).

The family of Zhao models is, however, not the only possible family of models with the characteristic behaviour (56a) and (56b). We can generate various new families of models with this behaviour using the concept of sigmoid functions. Sigmoid functions, generally denoted as $\mathcal{S}(x)$, are monotonically increasing functions that map the entire real domain onto a finite range, which is usually taken to be the unit interval $[0, 1]$. Moreover, sigmoid functions have a single inflection point, usually taken at $x = 0$, such that $\mathcal{S}(x)$ is convex for $x < 0$ and concave for $x > 0$. Sigmoid functions play an important role in

Table 2. Some of the most commonly used sigmoid functions.

Name	$\mathcal{S}(x)$
Logistic	$\frac{1}{1 + e^{-x}}$
Hyperbolic tangent	$\frac{1 + \tanh x}{2}$
Arctangent	$\frac{1}{2} \left(1 + \frac{2}{\pi} \arctan x \right)$
Error	$\frac{1 + \operatorname{erf} x}{2}$
Gompertz	$\exp(-e^{-x})$
Gudermannian	$\frac{1}{2} \left[1 + \frac{4}{\pi} \arctan(\tanh x) \right]$
Algebraic	$\frac{1}{2} \left(1 + \frac{x}{\sqrt{1 + x^2}} \right)$
Richards	$\frac{1}{(1 + \nu e^{-x})^{1/\nu}} \quad (\nu > 0)$

Notes. All these functions are normalised to have horizontal asymptotes at 0 and 1, and a single inflection point at $x = 0$.

the theory of population dynamics in a wide variety of scientific disciplines (e.g., Richards 1959; Tsoularis 2002; Banks 2013; Fornalski et al. 2020). Sigmoid function are also commonly used in artificial intelligence as activation functions in neural networks (Hayou et al. 2019; Hurbans 2020). Commonly used sigmoid functions include the logistic function, the arctangent function, and the Gompertz function; a number of possible options are listed in Table 2.

Now consider any sigmoid function $\mathcal{S}(x)$ and set

$$\gamma(r) = \gamma + (\beta - \gamma) \mathcal{S} \left(\alpha \ln \left(\frac{r}{r_b} \right) \right). \quad (57)$$

Interpreting this expression as the logarithmic density slope of a spherical model, we can find the density profile as

$$\rho(r) = \rho_c \left(\frac{r}{r_b} \right)^{-\gamma} \exp \left[-\frac{\beta - \gamma}{\alpha} \int_{-\infty}^{\alpha \ln(r/r_b)} \mathcal{S}(x) dx \right], \quad (58)$$

with ρ_c an arbitrary constant. Due to the properties of the sigmoid function, we immediately know that the model with this density profile behaves exactly as expressions ((55a) and (55b)). As we can apply the strategy above for any sigmoid function, we can easily generate various new classes of spherical models with this required asymptotic behaviour. As long as the integral in Eq. (58) can be evaluated analytically, this results in an analytical density profile.

The obvious first option is to consider the logistic function, one of the most popular sigmoid functions,

$$\mathcal{S}(x) = \frac{1}{1 + e^{-x}}. \quad (59)$$

This logistic function was originally introduced to describe the evolution of the Belgian population (Verhulst 1845), and is sometimes referred to in the literature as the sigmoid function. Combining Eqs. (58) and (59) leads to the following expressions for the logarithmic density slope,

$$\gamma(r) = \frac{\gamma + \beta (r/r_b)^\alpha}{1 + (r/r_b)^\alpha}, \quad (60)$$

and the density profile,

$$\rho(r) = \rho_c \left(\frac{r}{r_b} \right)^{-\gamma} \left[1 + \left(\frac{r}{r_b} \right)^\alpha \right]^{-\frac{\beta-\gamma}{\alpha}}. \quad (61)$$

This is nothing but the general expression of the density profile of the Zhao model.

5.2. Algebraic sigmoid density models

As an alternative to the logistic function, consider the algebraic sigmoid function,

$$\mathcal{S}(x) = \frac{1}{2} \left(1 + \frac{x}{\sqrt{1+x^2}} \right). \quad (62)$$

This choice leads to a new family of models with logarithmic density slope

$$\gamma(r) = \frac{\beta + \gamma}{2} + \frac{\beta - \gamma}{2} \frac{\alpha \ln(r/r_b)}{\sqrt{1 + \alpha^2 \ln^2(r/r_b)}}, \quad (63)$$

and density profile

$$\rho(r) = \rho_c \left(\frac{r}{r_b} \right)^{-\frac{\beta+\gamma}{2}} \exp \left[-\frac{\beta-\gamma}{2\alpha} \sqrt{1 + \alpha^2 \ln^2 \left(\frac{r}{r_b} \right)} \right]. \quad (64)$$

We have added this family of models to the SpheCow code as the `SigmoidDensityModel` class. The only member functions we had to implement were the density and its first and second derivatives. All other photometric and dynamical properties of this family of models can be evaluated without any further ado.

In Fig. 4 we compare the dynamical properties of a subset of the algebraic sigmoid models defined by the density profile (64) and the Zhao models. The different panels represent a wide variety of profiles, all of which are automatically calculated by SpheCow. The thick lines in this figure represent the models from the new family of algebraic sigmoid models. For all models, we have fixed $M_{\text{tot}} = r_b = 1$, $\alpha = 1$, and $\beta = 4$, and we have considered different values of the inner slope γ . The thin lines correspond to Zhao models with exactly the same parameters. Actually, the subfamily of Zhao models with $(\alpha, \beta) = (1, 4)$ is the well-known family of γ -models (Dehnen 1993; Tremaine et al. 1994).

It is clear that corresponding models from the two families have the same asymptotic behaviour in all quantities, as expected as they have the same asymptotic density profile by construction. The properties of the models of both families display a significant variety at small radii, depending on the parameter γ that characterises the inner density slope. For $\gamma = 0$ the models have a finite central density, surface density, potential, and isotropic velocity dispersion. For $0 < \gamma < 1$ both families of models show a shallow density cusp, but still a finite central surface density and finite potential well. The isotropic dispersion profile now tends to zero at small radii. Models with $1 < \gamma < 2$ show a similar behaviour, except for their surface density profile which diverges in the centre. Models with $\gamma > 2$ have a strong density and surface density cusp, an infinitely deep potential well, and diverging circular velocity and isotropic dispersion profiles.

While the asymptotic behaviour of the corresponding Zhao and algebraic sigmoid models is the same, there are some minor differences in the details. For a fixed set of parameters, the algebraic sigmoid models are characterised by a steeper transition

between the inner and outer profiles, as shown in the top left and top middle panels. This difference propagates to all other properties as well. Algebraic sigmoid models with small values of γ have a deeper potential well than the corresponding γ -models, and the opposite is true for large values of γ . Looking at the isotropic distribution function and differential energy distribution, the transition between the low and high binding energy parts is much sharper for the algebraic sigmoid models. As a result, the Zhao models have a larger number of stars with low binding energies, whereas the algebraic sigmoid models have more stars around the knee of the differential energy distribution. The specific choice of the sigmoid function does matter for the details of the dynamical structure.

Focusing on the panels on the third row of Fig. 4 we see that all algebraic sigmoid models shown have an isotropic distribution function that is a positive and monotonically increasing function of binding energy. This implies that these models are self-consistent and physical, and stable against radial and non-radial perturbations (Antonov 1962; Doremus et al. 1971; Binney & Tremaine 2008). We do note, however, that this conclusion cannot be generalised for all members of the family of algebraic sigmoid models, in a similar way as for the family of Zhao models (Baes & Camps 2021). If the parameter α that determines the sharpness of the transition between the inner and outer profiles grows larger, the distribution function will first develop a kink around $\mathcal{E} = \Psi(r_b)$. When α increases even more, the distribution function will become negative, making the models with an isotropic orbital structure unphysical. In the limit $\alpha \rightarrow \infty$, the models reduce to the broken power-law (BPL) models discussed by Du et al. (2020) and Baes & Camps (2021). These models are characterised by the simple density profile

$$\rho(r) = \begin{cases} \rho_c \left(\frac{r}{r_b} \right)^{-\gamma} & \text{if } r \leq r_b, \\ \rho_c \left(\frac{r}{r_b} \right)^{-\beta} & \text{if } r \geq r_b. \end{cases} \quad (65)$$

This density profile is indeed what we obtain if we take the limit $\alpha \rightarrow \infty$ in expression (64). The BPL models can also be seen as sigmoid models with the Heaviside step function

$$\mathcal{S}(x) = \Theta(x) \equiv \begin{cases} 0 & \text{if } x < 0, \\ \frac{1}{2} & \text{if } x = 0, \\ 1 & \text{if } x > 0, \end{cases} \quad (66)$$

as their defining (degenerate) sigmoid function.

Finally, it is interesting to compare the panels on the last two rows of Fig. 4. These rows show similar quantities, but the third row corresponds to an isotropic orbital structure, while the bottom row shows the same properties for an Osipkov-Merritt orbital structure with $r_a = 0.6$. At large \mathcal{E} or Q , the isotropic and Osipkov-Merritt distribution functions for any given set of parameters show a similar behaviour. This is not so surprising, as large binding energies correspond to small radii, and the Osipkov-Merritt models are isotropic in the central regions. The behaviour at small \mathcal{E} or Q , corresponding to large radii, is very different, with a different slope for both the distribution function and the differential energy distribution. The most interesting difference, however, relates to the behaviour at intermediate binding energies. The anisotropy generates a kink in the distribution function and the differential energy distribution, especially for models with small values of γ . For the algebraic sigmoid model with $\gamma = 0$, this kink is so strong that the distribution function and the differential energy distribution become

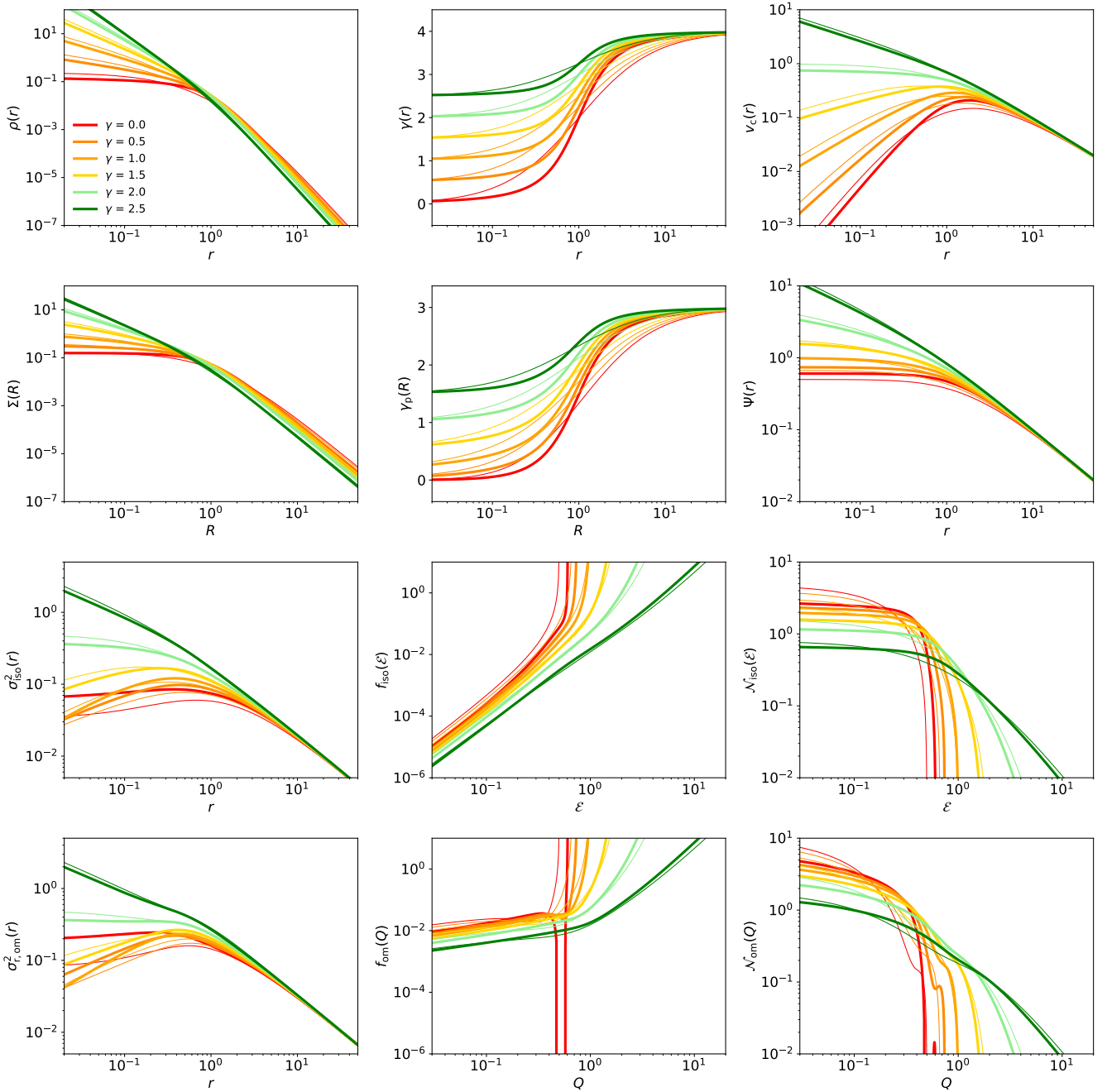


Fig. 4. Compilation of photometric and dynamical properties for two families of sigmoid models. The thick lines correspond to the new family of algebraic sigmoid models defined by the density profile (64). All models have $\alpha = 1$ and $\beta = 4$, and the different colours correspond to different values of the inner density slope γ , as indicated in the *top left panel*. The thin lines correspond to Zhao models with exactly the same parameters.

negative: this model cannot be supported by an Osipkov-Merritt orbital structure with $r_a = 0.6$. We note that the corresponding γ -model, that is the one with $\gamma = 0$ and $r_a = 0.6$, is also affected by the anisotropy, but the distribution function remains positive over the entire domain. This is in agreement with the results of [Carollo et al. \(1995\)](#), who found that the $\gamma = 0$ model can be supported by an Osipkov-Merritt orbital structure as long as $r_a > 0.44$.

5.3. Surface density sigmoid models

One of the useful characteristics of the SpheCow code is that it is essentially equally simple to generate models by means of a den-

sity profile or a surface density profile. Actually, for every model based on a certain density profile, we can immediately generate a counterpart with a surface brightness profile with a similar functional form, and vice versa. One well-known example of such a couple of twin models consists of the Einasto and Sérsic models. We note that the families of Einasto and Sérsic models have no connection to each other apart from a functional similarity between the density profile of the former and the surface density profile of the latter. In particular, Einasto models do not have a Sérsic surface density profile, and vice versa ([Dhar & Williams 2010](#); [Retana-Montenegro et al. 2012](#)).

In the same sense, it is also straightforward to generate new families of models in which the logarithmic slope of the surface

density profile has the shape of a sigmoid function. For any sigmoid function $\mathcal{S}(x)$, we set

$$\gamma_p(R) = \gamma + (\beta - \gamma) \mathcal{S}\left(\alpha \ln\left(\frac{R}{R_b}\right)\right), \quad (67)$$

resulting in a surface density profile

$$\Sigma(R) = \Sigma_c \left(\frac{R}{R_b}\right)^{-\gamma} \exp\left[-\frac{\beta - \gamma}{\alpha} \int_{-\infty}^{\alpha \ln(R/R_b)} \mathcal{S}(x) dx\right], \quad (68)$$

with a power-law behaviour at both small and large projected radii. If we take the logistic function as the standard sigmoid function, this results in the surface density profile,

$$\Sigma(R) = \Sigma_c \left(\frac{R}{R_b}\right)^{-\gamma} \left[1 + \left(\frac{R}{R_b}\right)^\alpha\right]^{-\frac{\beta - \gamma}{\alpha}}. \quad (69)$$

This surface density profile defines the family of Nuker models, frequently used to model the nuclear regions of galaxies (e.g., Lauer et al. 1995, 2005, 2007; Byun et al. 1996; Quillen et al. 2000; Rest et al. 2001; Laine et al. 2003; de Ruiter et al. 2005; Durret et al. 2019). Similar to the Einasto–Sérsic couple of models, the Zhao and Nuker models have no connection to each other except the similarity between the density profile (61) of the former and the surface density profile (69) of the latter (see also Zhao 1997).

Any other sigmoid function is possible as well, and will lead to a different family of models with slightly different photometric and dynamical properties. If we use the algebraic sigmoid function (62), we find

$$\Sigma(R) = \Sigma_c \left(\frac{R}{R_b}\right)^{-\frac{\beta + \gamma}{2}} \exp\left[-\frac{\beta - \gamma}{2\alpha} \sqrt{1 + \alpha^2 \ln^2\left(\frac{R}{R_b}\right)}\right]. \quad (70)$$

We have implemented the family of models defined by this surface density profile as the `SigmoidSurfaceDensityModel` class in the `SpheCow` code. This required just a few minor adaptations from the implementation of the `SigmoidDensityModel` class: the main adaptation is the required implementation of the third derivative of the surface density profile.

We do not provide an exhaustive overview or description of the dynamical properties of this new class of spherical models. It is not surprising that this general family of models shares many properties with the family of Nuker models, which has the same asymptotic behaviour. Just as in the previous subsection, there are minor differences in the surface density profile between corresponding models from both families, which propagate to all other properties. For a more detailed discussion of the Nuker models, we refer to Baes (2020), where we used a combination of analytical tools and a preliminary version of `SpheCow` to investigate the dynamical structure.

6. Discussion and summary

The goal of this work was to develop and describe a new software tool that can be used to set up a spherical dynamical model with either an arbitrary analytical density profile or an arbitrary analytical surface density profile as starting point. The ambition was that this code should be capable of automatically calculating the basic intrinsic and on-sky properties for any such model, as well as dynamical properties assuming either an isotropic or an Osipkov-Merritt type anisotropic orbital structure. With such a

code, it should be easy to replace the standard but not necessarily realistic analytical models for galaxies or dark matter haloes that are often used as starting points for theoretical or numerical investigations by more optimal alternatives.

The result of this work is `SpheCow`, a light-weight and flexible software code that has exactly the characteristics as described above. We have gathered all the relevant equations, and through some integral order rearrangements, derived some new expressions that make these calculations more simple. We have described our numerical integration strategy, which is based on a straightforward but very efficient Gauss-Legendre scheme. We have extensively validated `SpheCow` using a combination of comparisons to analytical calculations, and the calculation of inverse formulae. The code is publicly available as a set of C++ routines and as a Python module.

`SpheCow` contains readily usable implementations for many standard models, including the Plummer, Hernquist, NFW, Einasto, Sérsic, and Nuker models. The main strength of the code is, however, that it is designed to be easily extendable, in the sense that new models, defined by either an analytical density profile or an analytical surface density profile, can be added in a straightforward way. We demonstrate this by adding two new families of models: one with its logarithmic density slope and the other with its logarithmic surface density slope described by an algebraic sigmoid function. We do not investigate the properties of these models in full detail, but we show how straightforward it is to implement new models and how easy it is to subsequently explore their dynamical structure.

Apart from adding new models, the `SpheCow` code can be expanded in several other ways. One limitation of the current implementation is that only two orbital structures, isotropic and Osipkov-Merritt, are available. Other options are possible and probably desirable. In particular, neither the isotropic nor the Osipkov-Merritt case provides an accurate representation of the orbital structure of dark matter haloes as obtained from simulations (Cole & Lacey 1996; Colín et al. 2000; Mamon & Łokas 2005; Hansen & Moore 2006). Several other representations for the anisotropy profile have been proposed in the literature, which provide a more suitable fit to the results of N -body simulations (e.g., Carlberg et al. 1997; Diemand et al. 2004; Mamon & Łokas 2005; Tiret et al. 2007; Mamon & Boué 2010). Unfortunately, in the general case, the calculation of the distribution function for such general anisotropy profiles is cumbersome and ill-conditioned (Dejonghe 1986). There are, however, a number of special cases for which the inversion can be performed in a similar way as for the isotropic and Osipkov-Merritt models. The most obvious case are models with a constant anisotropy (Dejonghe 1986; An & Evans 2006b), and in particular the special cases of models populated only with radial orbits (Richstone & Tremaine 1984), or models with a constant radial anisotropy $\beta = \frac{1}{2}$ (Evans & An 2006). Other, more complex orbital structures for which the inversion can be performed (numerically) include the models by Cuddeford (1991) and Cuddeford & Louis (1995). We plan an implementation in `SpheCow` of dynamical models with some of these orbital structures as future work, and invite third parties to contribute to this effort.

This is not the only possible way in which `SpheCow` can easily be expanded. All models implemented at this stage are self-consistent self-gravitating models, that is, the density and gravitational potential are linked by Poisson's equation. It is also possible to drop this restriction and to include models consisting of two or more components. Such models can be useful, for example, to study the effect of a central supermassive

black hole and/or a dark matter halo on the stellar dynamical structure of a galaxy (e.g., Tremaine et al. 1994; Ciotti 1996; Ciotti et al. 1996, 2019; Baes & Dejonghe 2004; Baes et al. 2005; Ciotti & Ziaee Lorzad 2018).

We hope that this paper will inspire colleagues to use SpheCow, both to investigate the properties of existing models in detail and to explore models that have so far received less attention than they deserve.

Acknowledgements. M.B. and B.V. acknowledge financial support from the Belgian Science Policy Office (BELSPO) through the PRODEX project “SPICA-SKIRT: A far-infrared photometry and polarimetry simulation toolbox in preparation of the SPICA mission” (C4000128500). The authors thank the anonymous referee for a prompt and constructive referee report.

References

- Agnello, A., Evans, N. W., & Romanowsky, A. J. 2014, *MNRAS*, **442**, 3284
- Alcock, C., Allsman, R. A., Axelrod, T. S., et al. 1995, *ApJ*, **449**, 28
- Allaert, F., Gentile, G., & Baes, M. 2017, *A&A*, **605**, A55
- An, J. H., & Evans, N. W. 2006a, *AJ*, **131**, 782
- An, J. H., & Evans, N. W. 2006b, *ApJ*, **642**, 752
- Antonov, V. A. 1962, *Solution of the Problem of Stability of Stellar System* (Leningrad: Vestnik Leningradskogo Universiteta)
- Baes, M. 2020, *A&A*, **634**, A109
- Baes, M., & Camps, P. 2021, *MNRAS*, **503**, 2955
- Baes, M., & Ciotti, L. 2019a, *A&A*, **630**, A113
- Baes, M., & Ciotti, L. 2019b, *A&A*, **626**, A110
- Baes, M., & Dejonghe, H. 2002, *A&A*, **393**, 485
- Baes, M., & Dejonghe, H. 2004, *MNRAS*, **351**, 18
- Baes, M., & Gentile, G. 2011, *A&A*, **525**, A136
- Baes, M., & Van Hese, E. 2007, *A&A*, **471**, 419
- Baes, M., & Van Hese, E. 2011, *A&A*, **534**, A69
- Baes, M., Dejonghe, H., & Buyle, P. 2005, *A&A*, **432**, 411
- Banks, R. 2013, *Growth and Diffusion Phenomena: Mathematical Frameworks and Applications, Texts in Applied Mathematics* (Berlin, Heidelberg: Springer)
- Binney, J. 1982, *MNRAS*, **200**, 951
- Binney, J., & Mamon, G. A. 1982, *MNRAS*, **200**, 361
- Binney, J., & Tremaine, S. 2008, *Galactic Dynamics*, 2nd edn. (Princeton: Princeton University Press)
- Bovy, J. 2015, *ApJS*, **216**, 29
- Burkert, A. 1995, *ApJ*, **447**, L25
- Butsky, I., Macciò, A. V., Dutton, A. A., et al. 2016, *MNRAS*, **462**, 663
- Byun, Y. I., Grillmair, C. J., Faber, S. M., et al. 1996, *AJ*, **111**, 1889
- Cardone, V. F., Piedipalumbo, E., & Tortora, C. 2005, *MNRAS*, **358**, 1325
- Carlberg, R. G., Yee, H. K. C., Ellingson, E., et al. 1997, *ApJ*, **485**, L13
- Carollo, C. M., de Zeeuw, P. T., & van der Marel, R. P. 1995, *MNRAS*, **276**, 1131
- Ciotti, L. 1991, *A&A*, **249**, 99
- Ciotti, L. 1996, *ApJ*, **471**, 68
- Ciotti, L., & Ziaee Lorzad, A. 2018, *MNRAS*, **473**, 5476
- Ciotti, L., Lanzoni, B., & Renzini, A. 1996, *MNRAS*, **282**, 1
- Ciotti, L., Mancino, A., & Pellegrini, S. 2019, *MNRAS*, **490**, 2656
- Cole, S., & Lacey, C. 1996, *MNRAS*, **281**, 716
- Colín, P., Klypin, A. A., & Kravtsov, A. V. 2000, *ApJ*, **539**, 561
- Cuddeford, P. 1991, *MNRAS*, **253**, 414
- Cuddeford, P., & Louis, P. 1995, *MNRAS*, **275**, 1017
- de Ruiter, H. R., Parma, P., Capetti, A., et al. 2005, *A&A*, **439**, 487
- de Vaucouleurs, G. 1948, *Ann. Astrophys.*, **11**, 247
- de Zeeuw, T. 1985, *MNRAS*, **216**, 273
- Dehnen, W. 1993, *MNRAS*, **265**, 250
- Dejonghe, H. 1986, *Phys. Rep.*, **133**, 217
- Dejonghe, H. 1987, *MNRAS*, **224**, 13
- Dekel, A., Ishai, G., Dutton, A. A., & Maccio, A. V. 2017, *MNRAS*, **468**, 1005
- Dhar, B. K., & Williams, L. L. R. 2010, *MNRAS*, **405**, 340
- Di Cintio, A., Brook, C. B., Dutton, A. A., et al. 2014, *MNRAS*, **441**, 2986
- Diemand, J., Moore, B., & Stadel, J. 2004, *MNRAS*, **352**, 535
- Doremus, J.-P., Feix, M. R., & Baumann, G. 1971, *Phys. Rev. Lett.*, **26**, 725
- Du, W., Zhao, G.-B., Fan, Z., et al. 2020, *ApJ*, **892**, 62
- Durret, F., Tarricq, Y., Márquez, I., Ashkar, H., & Adami, C. 2019, *A&A*, **622**, A78
- Eddington, A. S. 1916, *MNRAS*, **76**, 572
- Einasto, J. 1965, *Trudy Astrofizicheskogo Instituta Alma-Ata*, **5**, 87
- Evans, N. W., & An, J. 2005, *MNRAS*, **360**, 492
- Evans, N. W., & An, J. H. 2006, *Phys. Rev. D*, **73**, 023524
- Evans, N. W., & Williams, A. A. 2014, *MNRAS*, **443**, 791
- Evans, N. W., An, J., Bowden, A., & Williams, A. A. 2015, *MNRAS*, **450**, 846
- Evans, N. W., O’Hare, C. A. J., & McCabe, C. 2019, *Phys. Rev. D*, **99**, 023012
- Fielder, C. E., Mao, Y.-Y., Zentner, A. R., et al. 2020, *MNRAS*, **499**, 2426
- Fornalski, K. W., Reszczyńska, J., Dobrzyński, L., Wysocki, P., & Janiak, M. K. 2020, *Acta Phys. Pol. A*, **138**, 854
- Freundlich, J., Dekel, A., Jiang, F., et al. 2020a, *MNRAS*, **491**, 4523
- Freundlich, J., Jiang, F., Dekel, A., et al. 2020b, *MNRAS*, **499**, 2912
- Graham, A. W., & Driver, S. P. 2005, *PASA*, **22**, 118
- Hague, P. R., & Wilkinson, M. I. 2013, *MNRAS*, **433**, 2314
- Hague, P. R., & Wilkinson, M. I. 2014, *MNRAS*, **443**, 3712
- Hague, P. R., & Wilkinson, M. I. 2015, *ApJ*, **800**, 15
- Hansen, S. H., & Moore, B. 2006, *New Astron.*, **11**, 333
- Hayashi, K., Chiba, M., & Ishiyama, T. 2020, *ApJ*, **904**, 45
- Hayou, S., Doucet, A., & Rousseau, J. 2019, in Proceedings of the 36th International Conference on Machine Learning, eds. K. Chaudhuri, & R. Salakhutdinov, *Proc. Mach. Learn. Res. (PMLR)*, **97**, 2672
- Hénon, M. 1959, *Ann. Astrophys.*, **22**, 126
- Hénon, M. 1960, *Ann. Astrophys.*, **23**, 474
- Hernquist, L. 1990, *ApJ*, **356**, 359
- Hurbans, R. 2020, *Grokking Artificial Intelligence Algorithms* (Shelter Island: Manning Publications)
- Jaffe, W. 1983, *MNRAS*, **202**, 995
- King, I. R. 1966, *AJ*, **71**, 64
- Laine, S., van der Marel, R. P., Lauer, T. R., et al. 2003, *AJ*, **125**, 478
- Lauer, T. R., Ajhar, E. A., Byun, Y. I., et al. 1995, *AJ*, **110**, 2622
- Lauer, T. R., Faber, S. M., Gebhardt, K., et al. 2005, *AJ*, **129**, 2138
- Lauer, T. R., Gebhardt, K., Faber, S. M., et al. 2007, *ApJ*, **664**, 226
- Lenze, D., Wagner, R., Rephaeli, Y., et al. 2012, *ApJ*, **752**, 141
- Lokas, E. L., & Mamon, G. A. 2001, *MNRAS*, **321**, 155
- Ludlow, A. D., Navarro, J. F., White, S. D. M., et al. 2011, *MNRAS*, **415**, 3895
- Mamon, G. A., & Boué, G. 2010, *MNRAS*, **401**, 2433
- Mamon, G. A., & Lokas, E. L. 2005, *MNRAS*, **363**, 705
- Mamon, G. A., Biviano, A., & Murante, G. 2010, *A&A*, **520**, A30
- May, A., & Binney, J. 1986, *MNRAS*, **221**, 13P
- Mazure, A., & Capelato, H. V. 2002, *A&A*, **383**, 384
- Merritt, D. 1985a, *AJ*, **90**, 1027
- Merritt, D. 1985b, *MNRAS*, **214**, 25P
- Michie, R. W. 1963, *MNRAS*, **125**, 127
- Michie, R. W., & Bodenheimer, P. H. 1963, *MNRAS*, **126**, 269
- Navarro, J. F., Frenk, C. S., & White, S. D. M. 1997, *ApJ*, **490**, 493
- Neureiter, B., Thomas, J., Saglia, R., et al. 2021, *MNRAS*, **500**, 1437
- Osipkov, L. P. 1979, *Pisma v Astronomicheskii Zhurnal*, **5**, 77
- Peñarrubia, J., Pontzen, A., Walker, M. G., & Koposov, S. E. 2012, *ApJ*, **759**, L42
- Plummer, H. C. 1911, *MNRAS*, **71**, 460
- Prugniel, P., & Simien, F. 1997, *A&A*, **321**, 111
- Quillen, A. C., Bower, G. A., & Stritzinger, M. 2000, *ApJS*, **128**, 85
- Rest, A., van den Bosch, F. C., Jaffe, W., et al. 2001, *AJ*, **121**, 2431
- Retana-Montenegro, E., van Hese, E., Gentile, G., Baes, M., & Frutos-Alfaro, F. 2012, *A&A*, **540**, A70
- Richards, F. J. 1959, *J. Exp. Bot.*, **10**, 290
- Richstone, D. O., & Tremaine, S. 1984, *ApJ*, **286**, 27
- Salucci, P., & Burkert, A. 2000, *ApJ*, **537**, L9
- Sérsic, J. L. 1968, *Atlas de Galaxias Australes* (Córdoba, Argentina: Observatorio Astronomico)
- Taylor, J. E., & Navarro, J. F. 2001, *ApJ*, **563**, 483
- Tiret, O., Combes, F., Angus, G. W., Famaey, B., & Zhao, H. S. 2007, *A&A*, **476**, L1
- Tremaine, S., Richstone, D. O., Byun, Y.-I., et al. 1994, *AJ*, **107**, 634
- Tsoularis, A. 2002, *Math. Biosci.*, **179**, 21
- Vasiliev, E. 2019, *MNRAS*, **482**, 1525
- Vasiliev, E., & Valluri, M. 2020, *ApJ*, **899**, 39
- Veltmann, U. I. K. 1979, *AZH*, **56**, 976
- Verhulst, P.-F. 1845, *Nouveaux Mémoires de l’Académie Royale des Sciences et Belles-Lettres de Bruxelles*, **18**, 1
- Wojtak, R., Gottlöber, S., & Klypin, A. 2013, *MNRAS*, **434**, 1576
- Zhao, H. 1996, *MNRAS*, **278**, 488
- Zhao, H. 1997, *MNRAS*, **287**, 525