Action-based modeling

Eugene Vasiliev

Oxford University

III Gaia Challenge workshop,
Barcelona, 2 September 2015

Plan of the talk

Definition of action/angle variables

Methods for computing action

How can actions be useful?

Software for the community

Definition of action/angle variables

- The instantaneous state of each star is given by 3 coordinates and 3 velocities: $\mathbf{x}$, $\mathbf{v}$.
  Not so helpful because all 6 variables change with time.
- An *orbit* in a given potential is characterized by its *integrals of motion*, which stay constant, and *location along the trajectory*.
- The integrals can be chosen in different ways; any function of them is also an intergral.
- Action/angle variables $\{\mathbf{J}, \boldsymbol{\theta}\}$ are the most natural way of describing the motion: from Hamilton's equations we have

  $\frac{dJ_i}{dt} = -\frac{\partial H}{\partial \theta_i} = 0$ (actions are integrals too), and

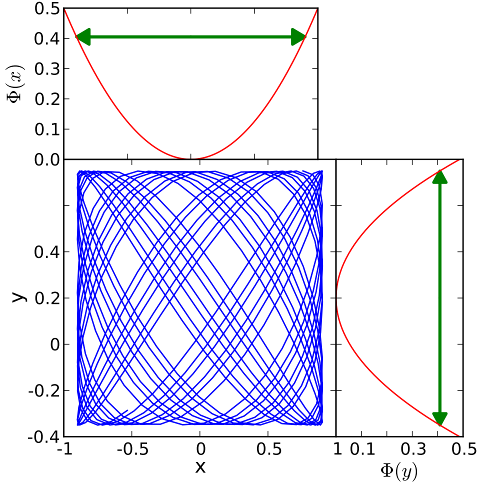
  $\frac{d\theta_i}{dt} = \frac{\partial H}{\partial J_i} \equiv \Omega_i$ (angles increase linearly with time);

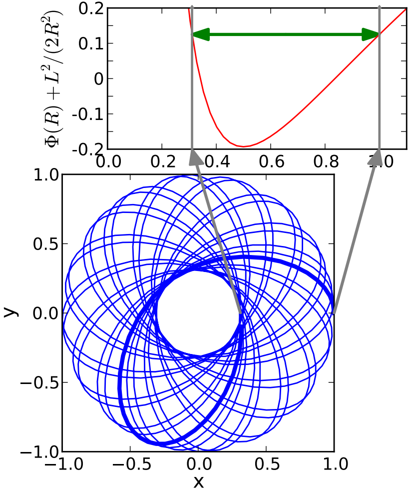
  here $H(\mathbf{J})$ is the Hamilonian and $\boldsymbol{\Omega}(\mathbf{J})$ are the frequencies.

Examples of action/angle variables

Actions are computed as $J = \dfrac{1}{2\pi}\oint \mathbf{p}\; d\mathbf{x}$, where $\mathbf{p}$ are canonically conjugate momenta for $\mathbf{x}$.

The meaning of the action/angle variables may vary for different classes of orbits.





Pros and cons of action/angle variables

+ Most natural description of motion (angles change linearly with time); once $\mathbf{J}$ and $\mathbf{\Omega}$ have been found, orbit computation is trivial.
+ Possible range for each action variable is $[0..\infty)$ or $(-\infty..\infty)$, independently of the other ones (unlike E and L, say).
+ Canonical coordinates: the volume of phase space $d^3x\; d^3v = d^3J\; d^3\theta$.
+ Actions are adiabatic invariants (are conserved under slow variation of potential); serve as a good starting point in perturbation theory.

— No general way of expressing the Hamiltonian $H \equiv \Phi(\mathbf{x}) + \frac{1}{2}\mathbf{v}^2$ in terms of actions (i.e., solving the Hamilton–Jacobi equation).

— Not easy to compute them in a general case.

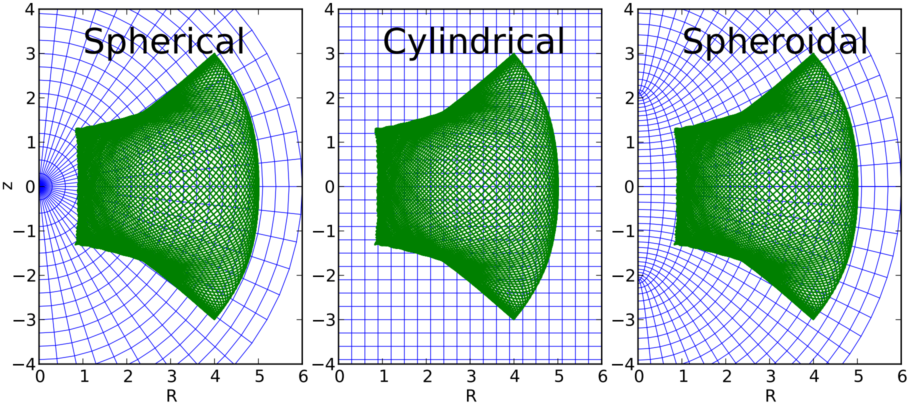
+ Efficient methods for conversion between $\{\mathbf{x}, \mathbf{v}\}$ and $\{\mathbf{J}, \boldsymbol{\theta}\}$ have been developed in the last few years.

“Classical” methods

- Spherical systems:
two of the actions can be taken to be the *azimuthal action* $J_\phi \equiv L_z$ and the *latitudinal action* $J_\vartheta \equiv L - |L_z|$;
the third one (the *radial action*) is given by a 1d quadrature:
$J_r = \frac{1}{\pi}\int_{r_{\rm min}}^{r_{\rm max}} dr\, \sqrt{2[E-\Phi(r)] - L^2/r^2}$, where $r_{\rm min}, r_{\rm max}$ are the peri- and apocentre radii.
Angles are given by 1d quadratures. For special cases (the isochrone potential, and its limiting cases – Kepler and harmonic potentials), these integrals are computed analytically.
Note: a related concept in celestial mechanics are the Delaunay variables.
- Flattened axisymmetric systems – the **epicyclic approximation**: motion close to the disc plane is nearly separable into the in-plane motion (J_ϕ and J_r as in the spherical case) and the vertical oscillation with a fixed energy E_z in a nearly harmonic potential (J_z).

State of the art: Stäckel fudge

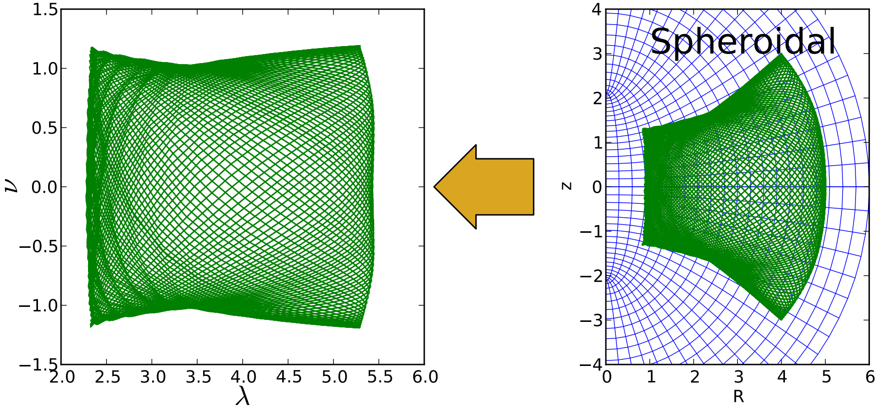
Fact: orbits in realistic axisymmetric galactic potentials are much better aligned with prolate spheroidal coordinates.



State of the art: Stäckel fudge

Fact: orbits in realistic axisymmetric galactic potentials are much better aligned with prolate spheroidal coordinates.

One may explore the assumption that the motion is separable in these coordinates (λ, ν).



Stäckel fudge (Binney 2012)

The most general form of potential that satisfies the separability condition is the Stäckel potential[1]: $\Phi(\lambda, \nu) = -\frac{f_1(\lambda) - f_2(\nu)}{\lambda - \nu}$.

The motion in λ and ν directions, with canonical momenta p_λ, p_ν, is governed by two separate equations:

$$2(\lambda - \Delta^2)\,\lambda\,p_\lambda^2 = \left[E - \frac{L_z^2}{2(\lambda - \Delta^2)}\right]\lambda - [I_3 + (\lambda - \nu)\Phi(\lambda, \nu)],$$

$$2(\nu - \Delta^2)\,\nu\,p_\nu^2 = \left[E - \frac{L_z^2}{2(\nu - \Delta^2)}\right]\nu - [I_3 + (\nu - \lambda)\Phi(\lambda, \nu)].$$

Under the approximation that the separation constant I_3 is indeed conserved along the orbit, this allows to compute the actions:

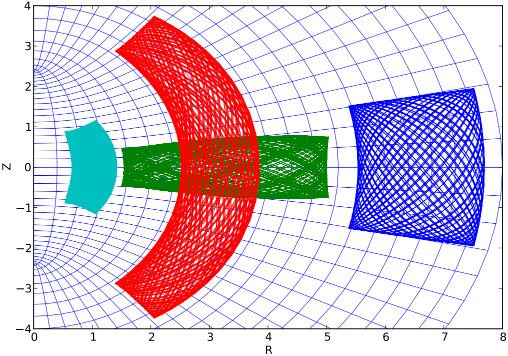
$J_\lambda = \frac{1}{\pi}\int_{\lambda_{\rm min}}^{\lambda_{\rm max}} p_\lambda\,d\lambda, \quad J_\nu = \frac{1}{\pi}\int_{\nu_{\rm min}}^{\nu_{\rm max}} p_\nu\,d\nu.$

[1] Note that the potential of the Perfect Ellipsoid (de Zeeuw 1985) is of the Stäckel form, but it is only one example of a much wider class of potentials.

Stäckel fudge in practice

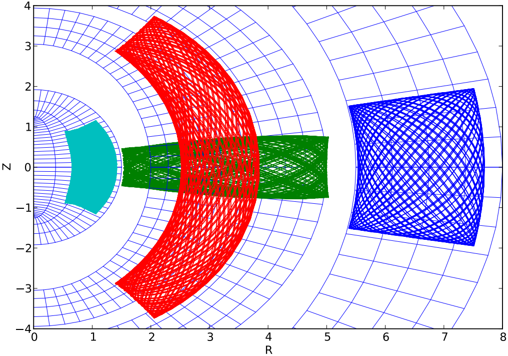
A rather flexible approximation: for each orbit, we have the freedom of using two functions $f_1(\lambda)$, $f_2(\nu)$ (directly evaluated from the actual potential $\Phi(R, z)$) to describe the motion in two independent directions.

These functions are different for each orbit (implicitly depend on E, L_z, I_3).

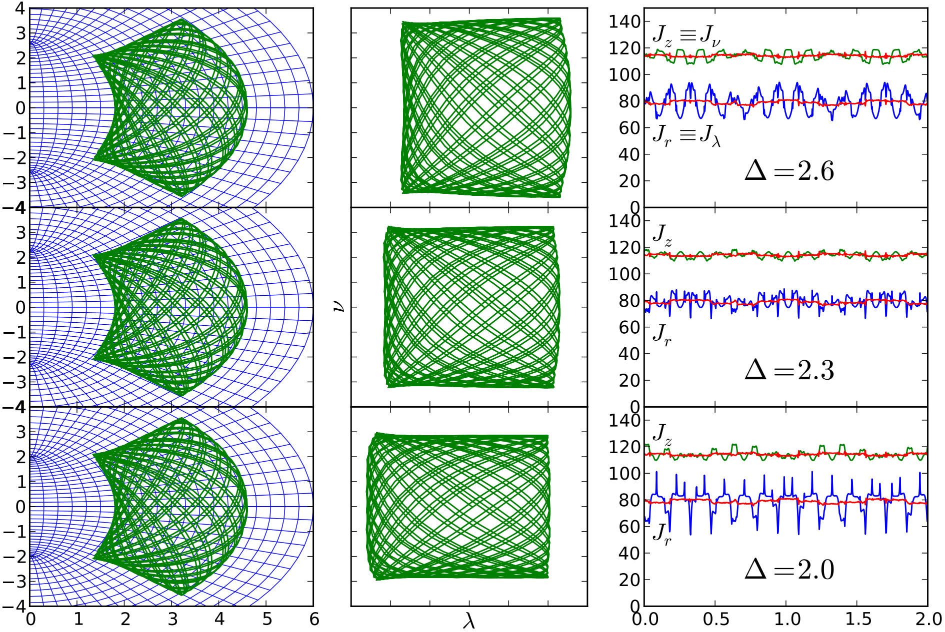


Stäckel fudge in practice

A rather flexible approximation: for each orbit, we have the freedom of using two functions $f_1(\lambda)$, $f_2(\nu)$ (directly evaluated from the actual potential $\Phi(R, z)$) to describe the motion in two independent directions.
These functions are different for each orbit (implicitly depend on E, L_z, I_3).
Moreover, we may choose the interfocal distance Δ of the auxiliary prolate spheroidal coordinate system for each orbit independently.



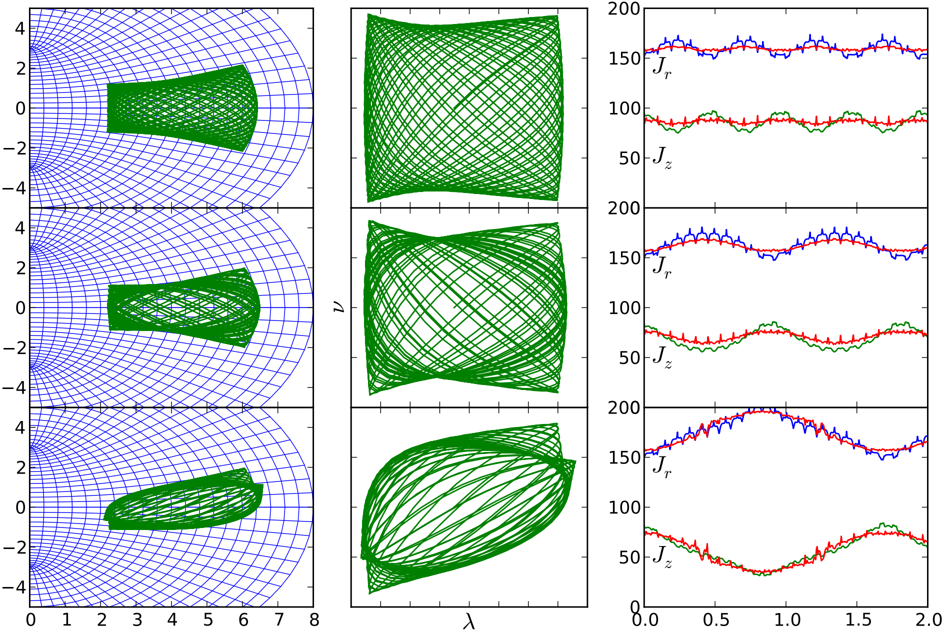
Importance of the interfocal distance parameter Δ



Other methods for determining the actions

- Methods with limited and uncontrollable accuracy:
  - Adiabatic approximation (correction for the energy of vertical motion in the epicyclic approach) [McMillan&Binney 2011, Schönrich&Binney 2012].
  - Fitting a Stäckel potential to a numerically-integrated orbit [Sanders 2012].
  - Triaxial Stäckel fudge [Sanders&Binney 2015].
- Methods based on canonical transformation between true $\{\mathbf{J}, \boldsymbol{\theta}\}$ and "toy" $\{\mathbf{J}^T, \boldsymbol{\theta}^T\}$ in some simple potential (e.g., isochrone), for which the mapping between position/velocity and action/angle coordinates is known [McGill&Binney 1990]. This transformation is described by a generating function $S(\mathbf{J}, \boldsymbol{\theta}^T)$, which can be expanded into Fourier series in $\boldsymbol{\theta}^T$; the accuracy of this approximation depends on the number of terms in the expansion.
  - Fitting a toy Hamiltonian to numerically integrated orbits [Fox 2012, Sanders&Binney 2014, Ueda et al.2014, Bovy 2014].
  - Using the inverse transformation – torus mapping, i.e., procedure for computing $\{\mathbf{x}, \mathbf{v}\}$ from $\{\mathbf{J}, \boldsymbol{\theta}\}$ [McMillan&Binney 2008] iteratively to improve the accuracy of actions [Sanders 2014].

Problems with (near-)resonant orbits

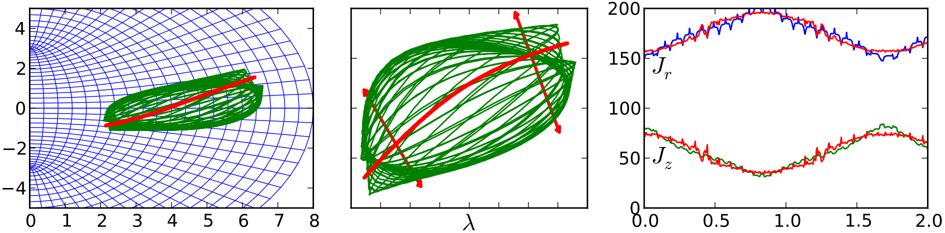


Problems with (near-)resonant orbits

For the resonant orbits the meaning of actions and angles is different from neighbouring non-resonant orbits (they describe the oscillation around a different parent closed orbit).

The canonical-transformation methods fail because of the problem of "small denominators" that breaks down the convergence of series expansion.

While the methods for dealing with resonant orbits in torus mapping have been introduced long time ago [Kaasalainen&Binney 1994, Kaasalainen 1995], they have not yet been implemented in practical applications.



Applications of action/angle variables

- Action-based distribution functions for disk and halo components [Ting et al.2013, Posti et al.2015, Williams&Evans 2015];
- Extended distribution functions that describe multiple stellar populations [Sanders&Binney 2015, Das&Binney in prep.];
- Radial migration – J_ϕ changes while J_r, J_z are ≈const [Sellwood&Binney 2002];
- Resonant diffusion formulated as perturbation theory in action-angle variables [Fouvry et al.2015];
- Adiabatic compression of halo as a process that conserves actions [Pontzen&Governato 2013, Piffl et al.2015];
- Self-consistent multi-component models [Piffl&Binney 2015];
- Tidal streams – extended in real space, but clustered in action space [Bovy 2014, Sanders&Binney 2014, Sanderson et al.2015];
- ...

The software for action-based modeling

- An extensive and versatile collection of gravitational potential models: analytic models such as double-exponential disc (using the efficient approach of Dehnen&Binney 1998), accurate and efficient general-purpose potential approximations for analytic density profiles and for N-body models.
- Powerful orbit analysis tools (fundamental frequencies, orbit classification, chaos detection techniques).
- Conversion between position/velocity and action/angle variables: Stäckel fudge, Torus modeling, orbit-based approaches.
- Action-based distribution functions; generation of N-body models and mock catalogues and determination of best-fit parameters of DF and potential taking into account the selection function.
- Self-consistent DF-based models (potential-density profile determined by DF).
- Efficient and carefully designed C++ implementation, Python interface, compatibility with other software such as galpy, pynbody.

(features already implemented but not yet incorporated int the library are in grey).

Summary

- Action/angle variables are a useful tool in galactic dynamics;
- There exist efficient methods for computing the actions in realistic galactic potentials;
- Constructing the DF-based models in terms of actions offers important advantages;
- "Reference implementation" available to the community.

  (see the presentation of software later today).

THANK YOU!

Presentation of the ABGal library: Action-based galaxy modeling framework[2]

- Computation of gravitational potential and forces;
- Orbit integration and analysis;
- Conversion between position/velocity and action/angle coordinates;
- Distribution functions (extended DF – age, metallicity, [α/Fe], ...);
- Selection functions;
- Framework for finding best-fit parameters of a model given the observational or mock data.
- Generation of mock data from DFs.

`https://github.com/GalacticDynamics-Oxford/ABGal`

[2] Provisional name; alternative suggestions are welcome!

Gravitational potentials

- Analytic density models are plenty; corresponding potentials are scarce or often expensive to compute (e.g., requires [multidimensional] integration);
- N-body models have self-consistent potential ready, but it's also expensive to compute (even with tree-code), and also too noisy;

Universal recipe?

- For halo-like components – spherical-harmonic expansion: $\Phi(r,\theta,\phi) = \sum_{l=0}^{l_\mathrm{max}} \sum_{m=-l}^{l} \Phi_{lm}(r) \times Y_l^m(\theta,\phi)$;
- For strongly flattened, disc-like components – Fourier expansion in ϕ with 2d interpolation of coefficients in R, z plane: $\Phi(R,z,\phi) = \sum_{m=-m_\mathrm{max}}^{m_\mathrm{max}} \Phi_m(R,z) \times \{\cos\, m\phi, \sin\, m\phi\}$.

Gravitational potentials

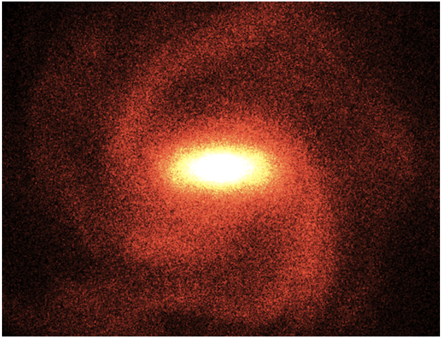
- + Provides smooth potential and forces suitable for orbit integration;
- + Second derivatives of potential for orbit analysis;
- + Coefficients may be computed either from an analytic density model, or from an N-body snapshot;
- + May vary the degree of symmetry (from spherical to triaxial and beyond);
- + Completely non-parametric;
- + Once initialized, force computation is very fast;
- — Suitable only for isolated well-centered systems with no substructure.

- + **Ready to use out of the box!**
  (C/C++ and Python interfaces; *NEMO*, *AMUSE* and *galpy* wrappers).

Gravitational potential approximations

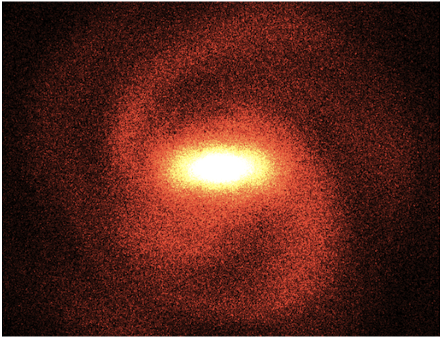
Real N-body model

(from Roca-Fabrega et al. 2013, 2014)



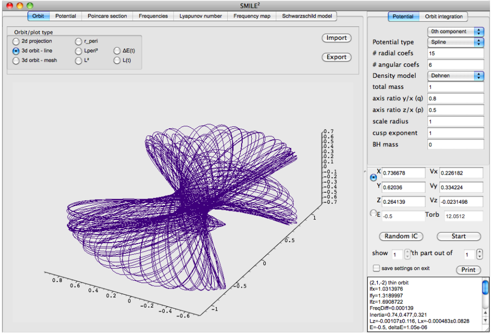
Potential approximation

(suitable for test-particle integrations, e.g. Romero-Gomez et al. 2011)



Orbit integration and analysis

- Interactive graphical interface;
- Orbit classification tools (e.g., Fourier analysis);
- Chaos detection methods (e.g., Lyapunov exponents);
- Tools for exploring the phase space (frequency maps, (Poincaré surfaces of section).



It can do quite a bit more,
including orbit-superposition models using the Schwarzschild's method.

`http://td.lpi.ru/~eugvas/smile/`

Action/angle finders

- From $\{\mathbf{x},\mathbf{v}\}$ to $\{\mathbf{J},\boldsymbol{\theta}\}$ – Stäckel fudge [Binney 2012]
(ABGal: more accurate and $\gtrsim$ 5x faster than all other implementations, including the original one);
- From $\{\mathbf{J},\boldsymbol{\theta}\}$ to $\{\mathbf{x},\mathbf{v}\}$ – Torus mapping [McMillan&Binney 2008]
(ABGal: 2x speedup, more flexible interface, plans to extend it for resonant orbits);

Demo programs:

- Test the accuracy of Stäckel fudge (variation of actions along numerically computed orbits);
- Fit an action-based DF to particles from an *N*-body simulation (with a self-consistently computed potential);
- Create an *N*-body model by sampling from a DF;
- Python example: compare the speed and accuracy of action finders between ABGal and galpy.

Selection function

$$P(S|\mathbf{u}) = p(S|l, b, s, v_{\parallel}, \mu_l, \mu_b, [\mathrm{Fe/H}])$$
$$= p(S|l, b)\, p(S|\mu_l, \mu_b)\, P(S|s, [\mathrm{Fe/H}])$$