# Action-based Galaxy Modelling Architecture

Eugene Vasiliev

Oxford University

Gaia Challenge, Stockholm, October 2016

https://github.com/GalacticDynamics-Oxford/Agama

## Have you ever experienced...

... a need to compute the potential from an arbitrary analytic density profile, or

... a frustration over your own Poisson solver being too slow, or

... an obsession with finding integrals of motion in a generic potential, or

... a desire to construct equilibrium dynamical models of galaxies, or

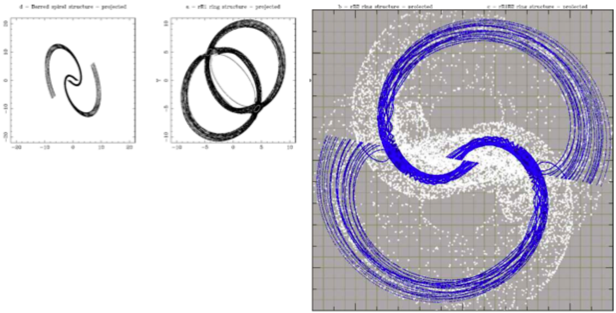
... a wish to analyze your clumsy *N*-body simulation with some analytic methods?...

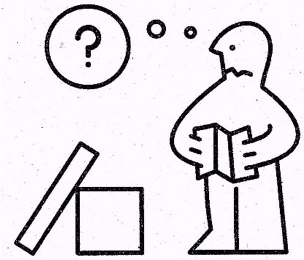
If so, then here is your prescription:

# Case study 1:

Invariant manifolds as building blocks for rings and spirals in barred galaxies

How to make it more realistic?

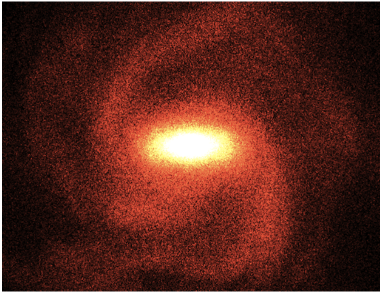




Response simulation in a fixed analytic potential

## Case study 1:

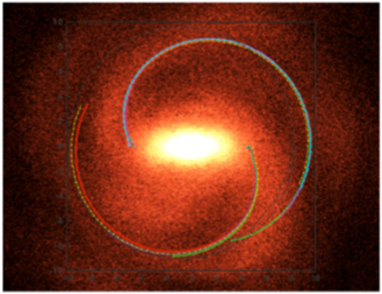
**1.**



**2.**

```
> import agama
> pot = agama.Potential \
      (file="snapshot.dat")
> orb = agama.orbit(pot, \
      initcond, time=100)
> plot(orb[:,0], orb[:,1])
```

**3.**



**4.**

## Potential solvers

- Two general-purpose potential approximations:

  1. Spherical harmonic expansion:

$$\Phi(r,\theta,\phi) = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \Phi_{lm}(r)\, Y_l^m(\theta,\phi).$$

interpolated functions

  2. Azimuthal harmonic expansion:

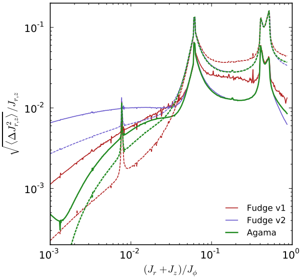
$$\Phi(R,z,\phi) = \sum_{m=-\infty}^{\infty} \Phi_m(R,z)\, \mathrm{e}^{im\phi}.$$

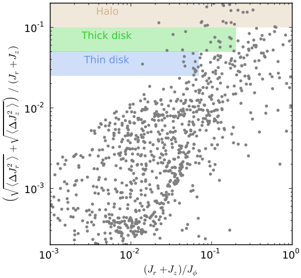
- Provide potential, force and its derivatives;
- Very accurate and computationally efficient;
- Can be constructed from any smooth density profile, or from an [expensive] user-defined potential routine, or from an $N$-body snapshot.

# Action/angle transformations

$\{\mathbf{x}, \mathbf{v}\} \Longleftrightarrow \{\mathbf{J}, \boldsymbol{\theta}\}$ for various classes of potentials:

- Spherical Isochrone model (golden classic);
- Arbitrary spherical potential:
  fast 2d interpolation over a grid in $\{E, L\}$;
- Axisymmetric potentials:
  Stäckel approximation ($\Rightarrow$), torus mapping ($\Leftarrow$);
  (3d interpolation over $\{E, L_z, I_3\}$,
  two-way torus mapping,
  triaxial potentials –
  work in progress).



## Action-based distribution functions

Flexible $f(\mathbf{J})$ for disk-like and spheroidal components.

Advantages of using actions:

- Same expression valid in any gravitational potential;
- Adiabatically conserved $f \implies$ may study the response of dark matter halo to disk formation, etc.
- Easy to construct multi-component models;
- Natural starting point for perturbation theory;
- May use full information about velocity distribution, not just first and second moments (applies to any DF-based model) $\implies$ better constraints on parameters of physically valid models;
- Possibility of non-parametric reconstruction of DF from an $N$-body snapshot (so far only for spherical isotropic models).

## Self-consistent models

- Single- and multi-component models with explicitly known $f(\mathbf{J})$
  (iterative solution for the potential/density corresponding to the given $f$);
- Models with non-parametric $f(\mathbf{J})$
  (variation of the Schwarzschild method, but with a smooth DF);
- Straightforward to perturb a model, or define a metric in the model space;
- May create $N$-body realizations of these models;
- A single model with reasonable resolution can be constructed in a few seconds to a few minutes.

## The AGAMA library

- Written in C++, with great attention to computational efficiency and numerical accuracy;
- Well documented – both in-code (Doxygen) and in a readme file (60 pages so far...);
- Many example programs and internal tests illustrating various usage aspects;
- Python interface for a large subset of its features (cool!);
- Fortran interface for potential solvers (spooky!);
- Plugins for Galpy, Amuse, Nemo.

https://github.com/GalacticDynamics-Oxford/Agama