Star Generator (demo ver.210120)

sarchaan

January 2021

1 Descriptions

1.1 Settings

- IMF: Kroupa(2001)
- Density profile: uniform density (for test)
- Velocity: Gaussian distribution, peaked at Keplerian velocities (guess)

1.2 Schematic procedures

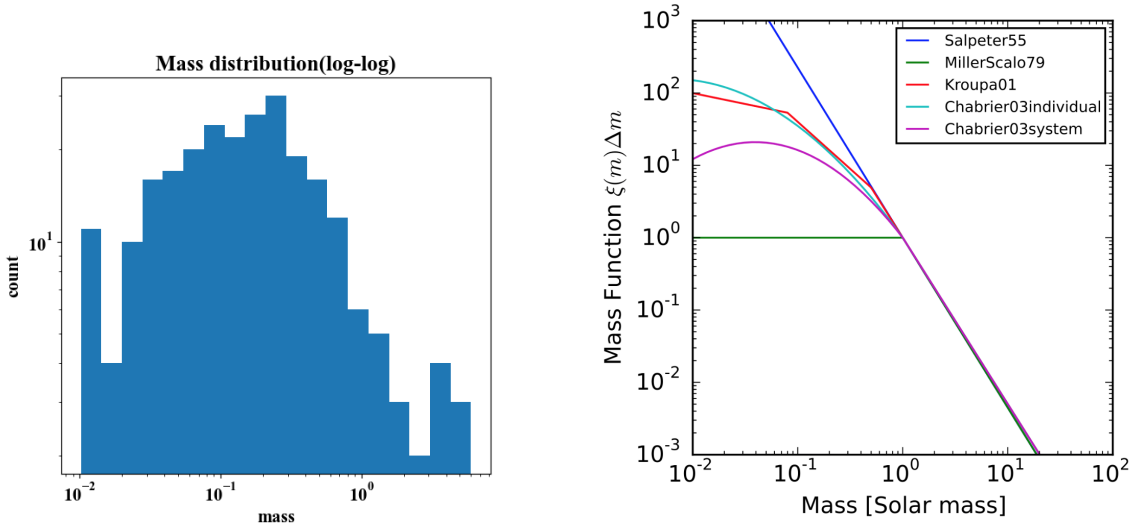
- Mass
  1. draw a cdf(cumulative distribution function) from the given IMF
  2. pick a random number u in [0,1)
  3. solve $u = \mathrm{cdf}(m)$ for m
- position
  1. draw a cdf from the given density profile
  2. same as above
  3. same as above
- velocity
  1. $v = \sqrt{\frac{GM_{enc}}{r}}$
  2. $v_t = \mathrm{Gaussian}(v, v/\sqrt{2})$
  3. $v_r = \mathrm{Gaussian}(0, v/\sqrt{2})$
  4. direction of v_t(tangential velocity) is uniformly random

2 Demo run results

- Cluster mass: 10^2 $M_\odot$ (given setting)
- Cluster size: 3.0 pc (given setting)
- ...Running...
- Running time: 86 seconds (!)
- Number of generated stars: 252
- Generated mass: 100.193283546998 $M_\odot$

2.1 Mass

The sampled mass distribution(Fig 1) is satisfactory overall, but seems to lack lightweight stars.



(a) Sampled mass(solar mass) distribution.

(b) Desired mass distribution. See Kroupa01(red)[1]

Figure 1: Mass distribution

2.2 r-v relation

It shows a Gaussian-distributed rigid-body-like rotation curve, which is natural from the fact that we used the uniform density profile.

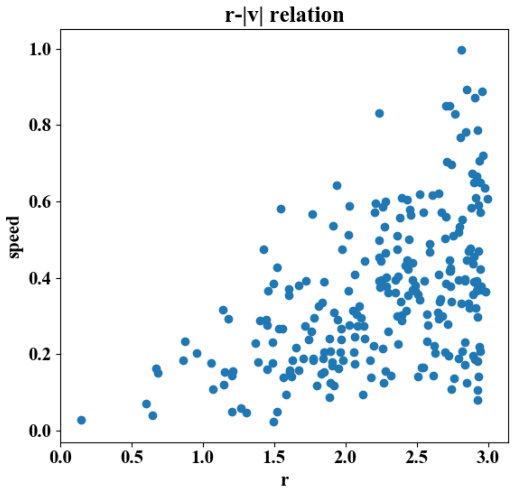


Figure 2: r-v plot. v(arbitrary unit) grows linearly with r(pc). (cf. galactic center)

3 Discussion

We must fix:

- The program is very slow. It would take 10 years to generate 10^9 stars. Possible seeds: (a) several random samplings inside a loop (b) several heavy symbolic functions like solve() or integrate() inside a loop. Possible solutions: (a) replace heavy functions with more primitive functions (b) use mpi (c) use supercomputer
- Another concern is that the sampled mass distribution seems somewhat bottom-light. It looks more like Chabrier(2003), not Kroupa(2001). Can't figure out why. We may test this with a larger sample.

In the next version we may try:

- different mass distribution e.g. $\rho \sim r^{-n}$
- more appropriate velocity distribution

We may refer to GalIC, which uses iterative methods to generate more realistic initial condtions.

References

[1] Johannes Buchner. Plot of various initial mass functions. URL `https://en.wikipedia.org/wiki/Initial_mass_function`.