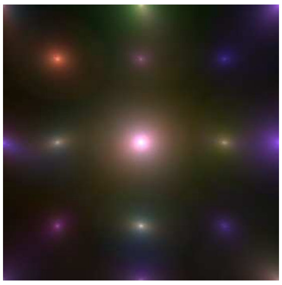
RealGal version 0.3

13th September 2005



Abstract

The purpose of the software RealGal is to generate "realistic" cosmological movies of galaxies via raytracing, while relying (transparently) on public domain softwares: pov-ray, yorick, mpeg_encode. It makes use of a library of (automatically generated) 3D-icons in df3 format (which can be replaced by user-provided "improved" cubes). It also requires knowing (or guessing) the position, luminosity, age and size of the objects. The ray tracing is carried within povray, hence it has the flexibility and limitation of this software. It may run in parallel, using the mpi version of pov-ray. This document describes the main features of RealGal, together with a rapid tutorial a a wish list for future features. It is a component of the HORIZON project, and inherits its copyrights and distribution policy.

1 Introduction

While cosmological simulations provide astronomers with direct estimates of particles, it remains computationnaly expensive to produce realistic snapshots which mimic both the large scale structures and the small galactic scales. Realgal intents to produce "nice looking" movies from simple catalogs of points (extracted from simulations).

1.1 Basic principle

The basic principle underlying this software is illustrated in the figure below; our policy is to replace points with properties in a catalog by hyper icons (*i.e.*

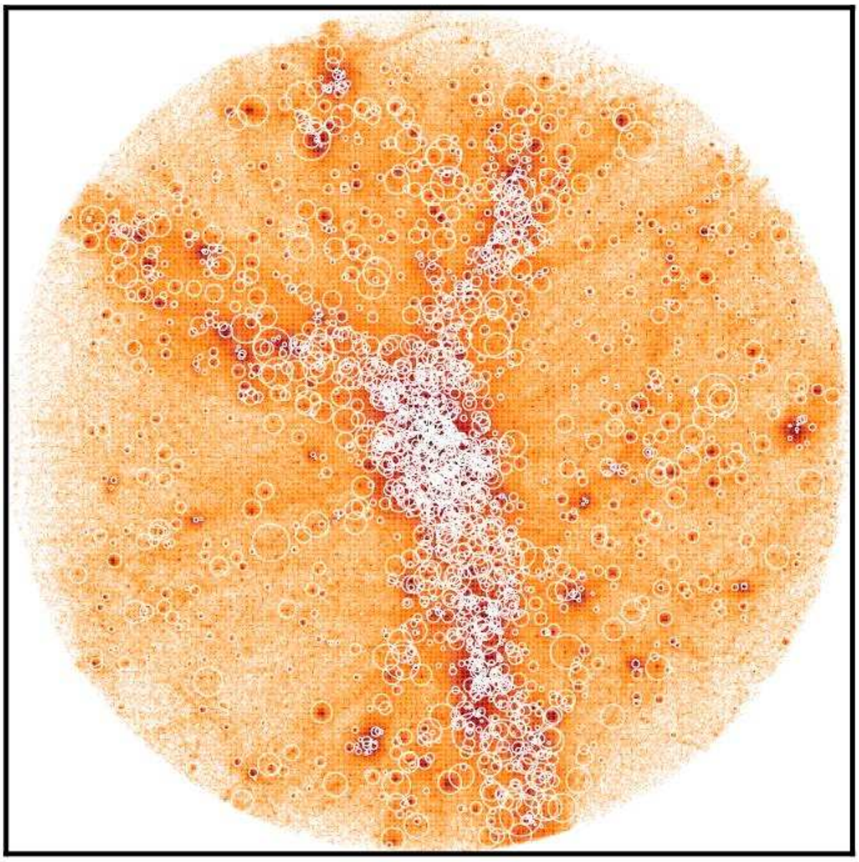
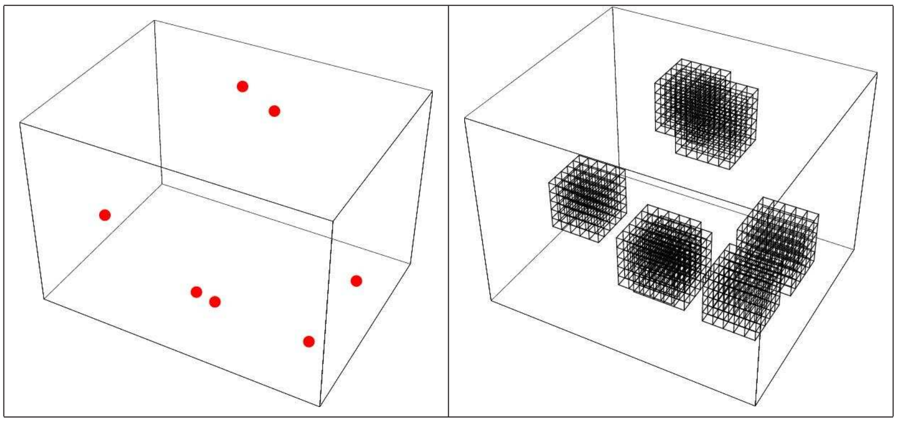


Figure 1: initial density file (a cluster of 10 Mpc extracted from a 512^3dark matter simulation) together with the identified *adaphop* substructures, which are feeded to realgal.



df3 cubes) at the corresponding position with possibly higher resolution (if the hyper icon is close to the camera path) and then rely on povray to do the ray tracing. It implements practically the concept of multiscale raytracing, in the spirit of a static (admitedly somewhat trivial) AMR code.

1.2 Quick start

1.2.1 Installation ...

The package can be downloaded as a gziped tar ball; `realgal-0.2.tar.gz` at the following link realgal `ftp://realgal-02.tgz`.

The package corresponds basically to one script, realgal, which takes at least one input, a catalog or a flag.

The installation is quite straightforward:

1) Setup the environmental variable REALGAL to this directory; e.g. (in tcsh) `setenv REALGAL ‘pwd‘`

2) If required, install yorick `http://www.maumae.net/yorick/doc/index.php`

3) If required, install povray http://www.povray.org

4) If required, install meg_encode `http://bmrc.berkeley.edu/ftp/pub/multimedia/mpeg/encode/mpeg_encode-1.5b-src.tar.gz`

5) Edit the first line of realgal so that it points to the yorick binary (itou for adapthop2cat if required).

6) possibly rename the `/usr/local/etc/povray.conf` file to something else. ????

1.2.2 First run

The easiest way to get started, once the software is installed is to type:

```
realgal -h;
```

to get an idea of what the flags are, and then

```
realgal -d;
```

This should make your first movie (in, by default `$REALGAL/output/mymovie/mymovie.mpg`) ! You may then, say, increase the size of the image (`-s` or `-size`), and change the number of frames (`-f` or `-frames`):

```
realgal -s 200 -f 1;
```

Beyond this point you may want to specify your own catalog;

```
realgal -m <myMovie> -c <myCatalog.cat> -s 50 -f 1;
```

where `<myMovie> and <myCatalog.cat>` correspond to your own name for the movie and catalog file (see below, Sec. 1.3). This time the movie file and corresponding files (see below) will be put in `$REALGAL/output/<myMovie>/<myMovie>.mpg`.

1.2.3 Customisation

Probably the best strategy is to first run the program in sketch mode at a low number of frames. (possibly one only), and set the `trav_scale`, (and possibly `trav_rot`) so that the view includes the desired objects at all time. For instance:

```
realgal --movie Grid --quality sketch -t ./input/line.trav
-c ./input/grid3x3x3.cat -s 200 -v 1 -f 1 -trav_scale 1.5
```

You may also change the time corresponding to the snapshot to check that your movie is still properly rendered at some later time:

```
realgal -m Grid -q sketch -t ./input/line.trav -c ./input/grid3x3x3.cat
-s 200 -v 1 -i 0.9 -f 1 -trav_scale 1.5
```

Once the setting is satisfactory, you may launch the production of the movie, in either low, med or high quality. Its probably best to first do a test run at low resolution and small image size

```
realgal -m Grid -q low -t ./input/line.trav -c ./input/grid3x3x3.cat
-s 50 -v 3 -f 10 -trav_scale 1.5
```

to check that everything is working according to expectations. You might want to adapt the global brightness of the film via the `brightness` flag.

1.3 Input-output

Inputs

1. position catalog at a given redshift .

- x, y, z, galactic type, color,luminosity

1. travelling file

- x, y, z of splines defining the travelling
- pointing of camera

Outputs

(in subdirectory `outputs/name_of_movie/`)

- mpeg file : `name_of_movie.mpg`

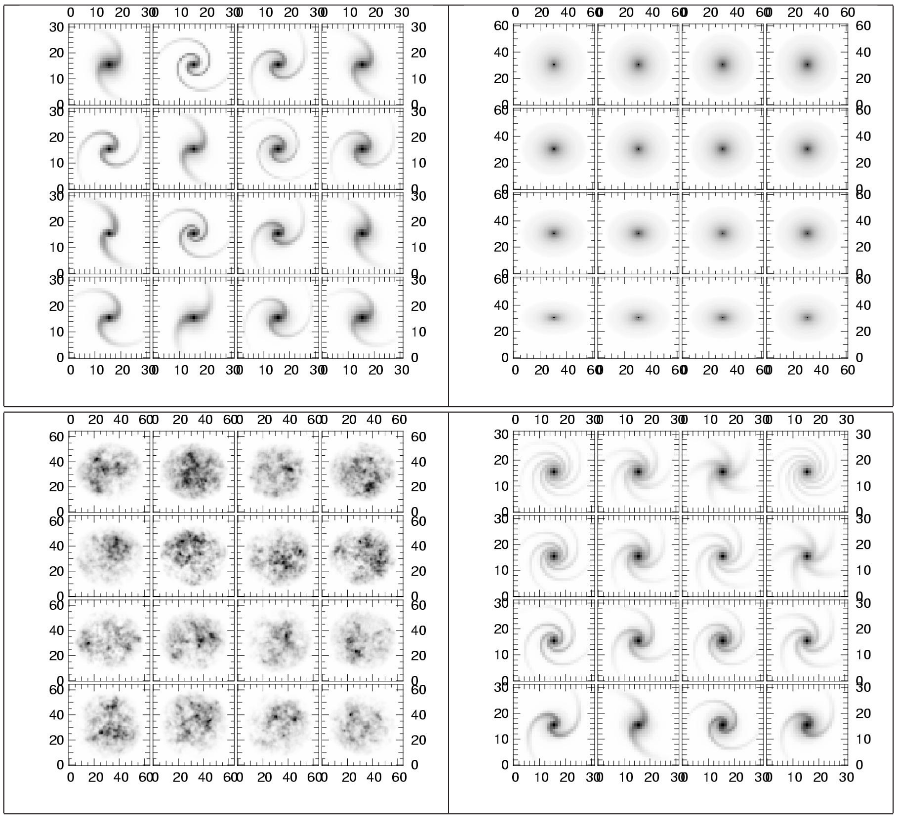


Figure 2: Library of hyper icons: from *left* to *right* and *top* to *bottom*: barred spirals, ellipticals, irregulars, spirals

- `name_of_movie.exe` to (re)generate .ppm (for relaunch)
- `name_of_movie.mpeg_encode_in` to (re)generate the .mpg (itou)
- the snapshots using the format `name_of_movie-snap-####.ppm` (view via display)
- the travelling used for the movie
- a copy of the catalog `name_of_movie.mpg`

Intermediate by-products

1. library of galaxies in df3 formats.

- Ellipticals
- Spirals
- Irregulars
- Barred galaxies
- Merging
- High resolution galaxies

1.3.1 Parameter files

the travelling file contains a keyword, either `fixed` or `ahead` which specifies if the camera should be pointing in front of the travelling or towards a fixed direction. In the latter case, the direction, x, y, z is given at the end of the file.

1.3.2 Flags

Realgal takes the following flags

- -h, –help [void] Display the help message.
- -d, –debug [void] Toggle on debug messages.
- -v, –verbosity [int] Verbosity level (from 0 to 4).
- -m, –movie [string] Movie directory name (in output).
- -c, –catalog [string] Catalog file name.
- -t, –travelling [string] Travelling file name.
- -n, –no-display [void] Do not display images during the rendering.
- -f, –frames [int] Number of frames (>=12 for movie production).

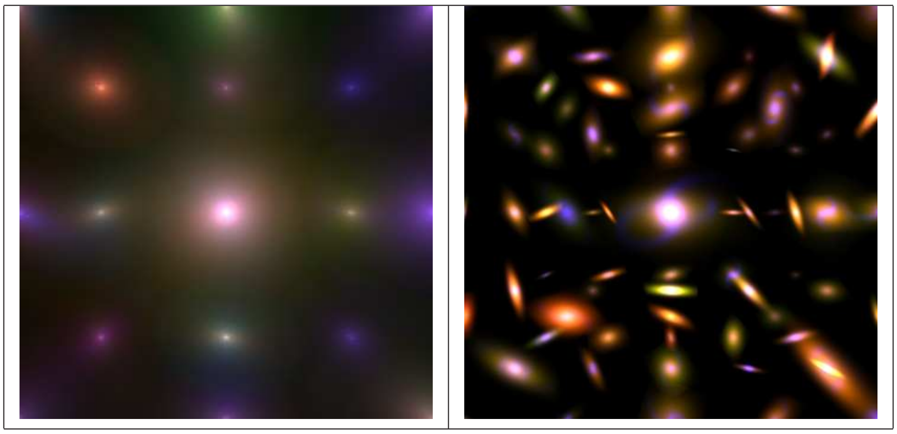


Figure 3: *left*: ellipticals and *right*: barred spirals on a regular grid as a testcase (produced by *realgal -c input/grid5x5x5.cat -t input/spiral-fixed.trav -b 1 -s 400 -q low -f 1*)

- -i, –initial [float] Initial frame (>0 & <=1).
- -b, –brightness [float] Brightness.
- -k, –keep [void] Keep images.
- -s, –size [int] Image size.
- -q, –quality [sketch, low, med or high] Image quality.
- –rotation-trav [float, float, float] Travelling rotation.
- –scale-trav [float] Scale the travelling.
- –scale-cat [float] Scale the catalog. If 0 willl rescale according to the rms. If negative, only rescales the size of objects.
- –threshold [float] Distance to the camera after which galaxies are simplified.

1.3.3 Travelling and Galactic catalogs

All default input files are stored in `$REALGAL/input/`
in particular the following input catalogs and travelling are provided:
grid1x1x1.cat : one galaxy at the center,
grid2x2x2.cat: a cubic grid 2x2,
grid3x3x3.cat: a cubic grid 3x3x3,
grid5x5x5.cat: a cubic grid 5x5x5,
rand500.cat: 500 random galaxies,

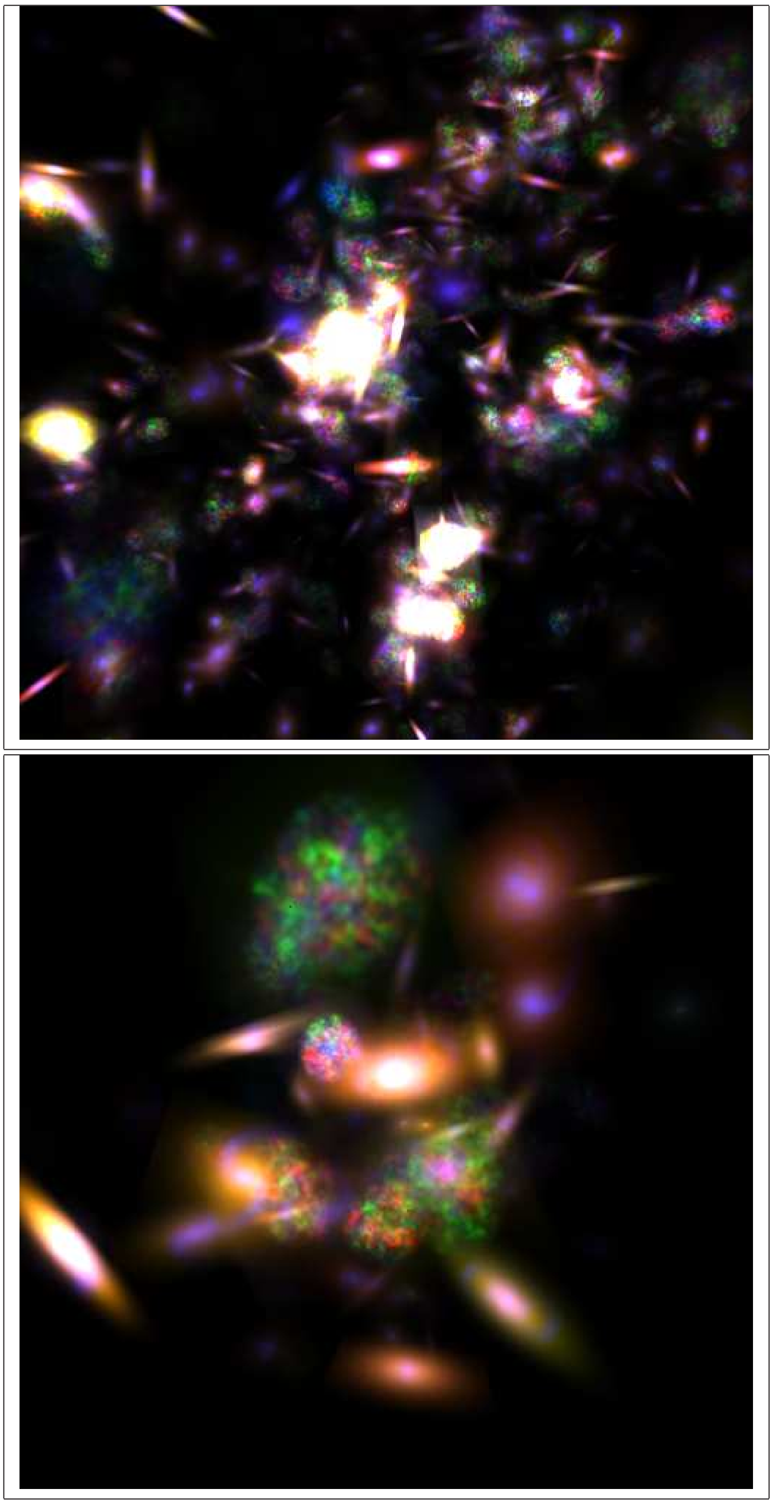


Figure 4: *up*: global cluster view; *d8wn*: close up of the cluster in Fig.1.

cluster.cat: a cluster (in fact the cluster depicted in Fig. 1),
line.trav: a linear travelling towards the center,
line-through.trav: a linear travelling towards and through the center,
circle-ahead.trav: a circular travelling around the center pointing ahead,
circle.trav: a circular travelling around the center pointing towards the center,
spiral-fixed.trav: a spiralling travelling pointing towards the center,
spiral-ahead.trav: a spiralling travelling pointing ahead.

For instance, the *line.trav* configuration file contains

```
line ← the name of the travelling
-2 -1 -0.5 ← the x coordinates of the spline
0 0 0← the y coordinates of the spline
0 0 0← the z coordinates of the spline
1 ← the number of opening angle of the camera
67.3← the angle of the camera
1 ← the number of pointing of the camera
2.1 0 0.← the coordinates of the pointing direction
```

This directory also includes the `povray.ini` file and the `mpeg_encode_ref` file which specify the default beheaviour of povray and mpeg_encode respectively.

2 Possible extensions

1. Travelling with splines for viewpoint and opening angle.
2. Realistic galaxies + mergers (FINISHED)
3. Realistic colors: realistic filters à la image_maker.
4. Improve dramatically the hyper icons (ALMOST FINISHED; pbs with irregulars)
5. Add dust (absorption...)
6. Add halo ? and subclumps.
7. Deal with the IGM
8. Add the gaz; diffuse density; global df3 corresponding to the underlying say,gaz density
9. If distance to camera below threshold use high resolution images ? (ALMOST FINISHED)
10. Add sub structures in the disks using GRF (FINISHED)
11. Add high resolution galaxies (FINISHED)

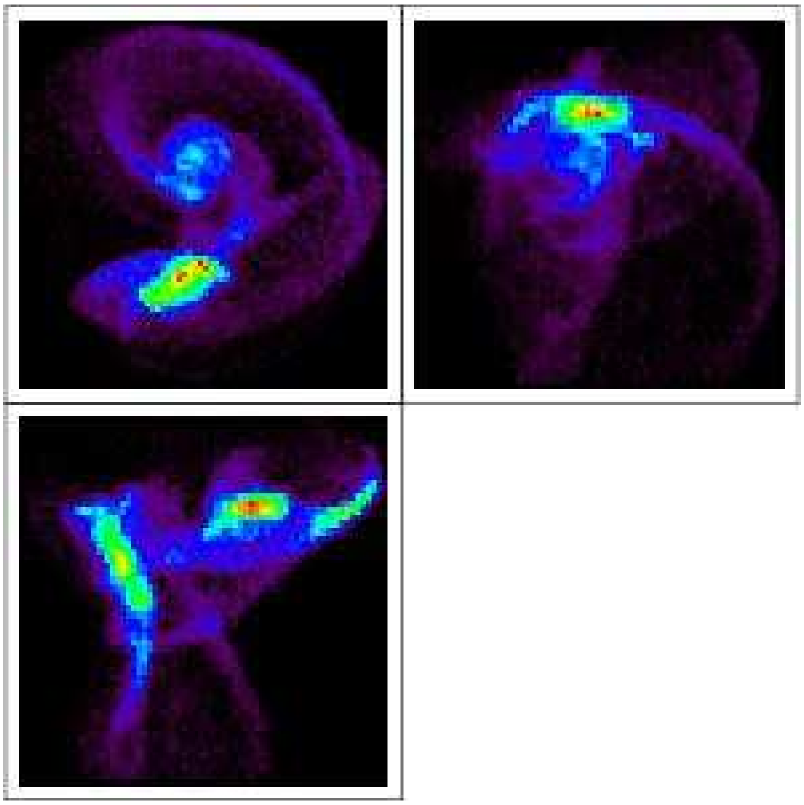


Figure 5: a merger 3D icon computed with gadget and magalie

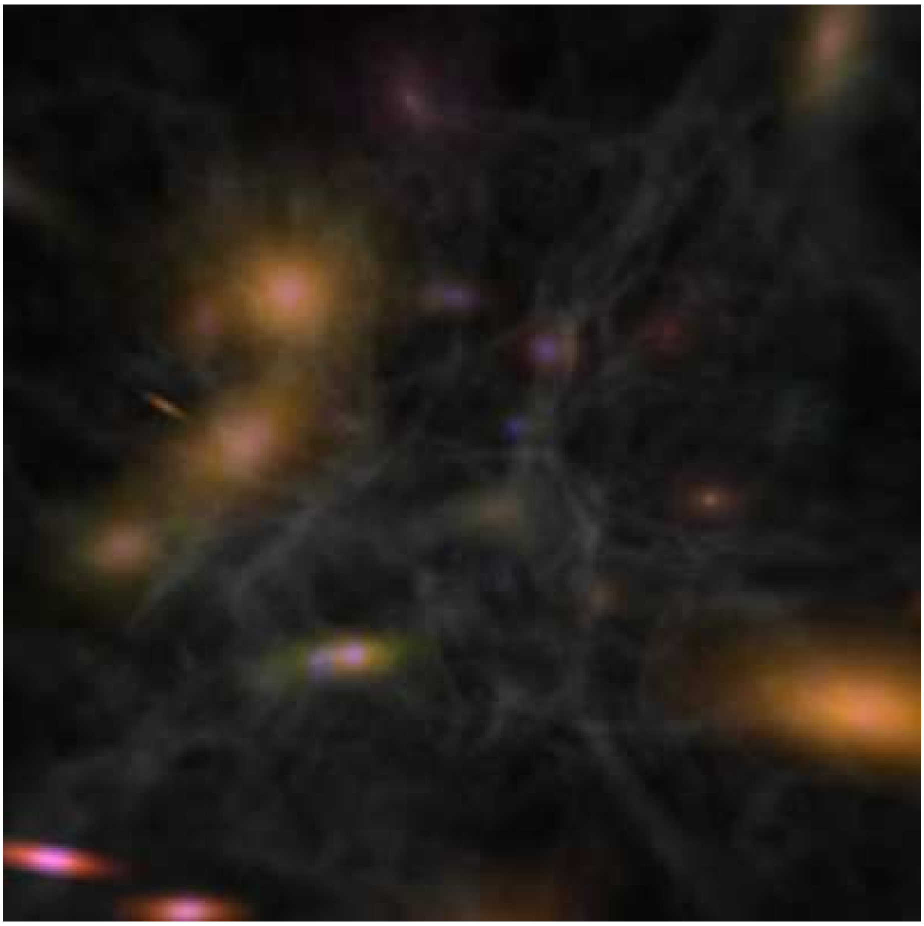


Figure 6: exemple of possible diffuse component, mixed with catalog objects.

12. Investigate mpi-povray; should be ok but pointless since no load balancing (wrote mp_source)
13. modify the script so that everything is relative to "$REALGAL"?
14. replace export to mpeg_encode with yorick mpeg implementation ?

3 Known bugs

- The nodisplay flag is disfunctionnal. The retain images flag is mandatory.
- If more than 256 galaxies lay along the same line of sight, this cause povray to crash. This can be dealt with by reducing the size of galaxies when the field becomes too crowded.

4 Concluding remarks

Let us first emphasize that the core of RealGal is independant of the type of 3D icons used; these could be extracted from, say AMR simulations

The underlying engine behind RealGal is yorick, an interpreter, and as such realgal can also be used directly via yorick for those so inclined (in fact the authors strongly recommand yorick as a very versatile scientific tool). Look at lib/yorick/realgal.i to discover the underlying yorick functions, described briefly in appendix A, starting with MakeMovie. We also provide for the script, adapthop2cat which converts adapthop node files into catalog input files for realgal. Note that adapthop2cat will keep track of the size of the identified substructures, fix the age of the galaxy according to the local density and fix their size according to the estimated size of the clumps.

Acknowledgments *We thank D. Aubert, E. Thiébaut, R. Teyssier and S. Colombi for inputs, D. Munro for developping* yorick, *and the* povray *team for doing the real work!*

A Yorick Functions

Yorick Documentation for functions, variables, and structuresdefined in file lib/yorick/realgal.i

- `adaphop2cat`

```
convert adaphop clusters to catalog;
   if pos_only =1 assumes only positions are given
   as a ymwrite file
   EXAMPLE
   adaphop2cat("tmp/hal2.nod","input/cluster2.cat")
   adaphop2cat("pos.asc","input/amas.cat",pos_only=1);
```

- `interp_travel`

```
interp traveling along t array or scalar in [0,1];
   rot= applies a rotation
   if dir=1 puts in vel the orientation of the normal to the trajectory
   extra_pts= adds some points before applying the spline interpolation
   (for the purpose of say a larger object)
```

- `makeMovie`

```
generates .mpg given a catalog or a array of galaxies and a traveling file or structure.
   FLAGS:
   nsnap = number of snapshots
   size of image
   run if set, produces the images and the movie as well
   verb 1,2,3 verbose level
   res, sketch,low,med,high resolution of images
   brightness how bright the galaxies are
   rot_trav = rotation of traveling
   scale_trav = scaling of traveling
   nodisplay = avoids X output
```

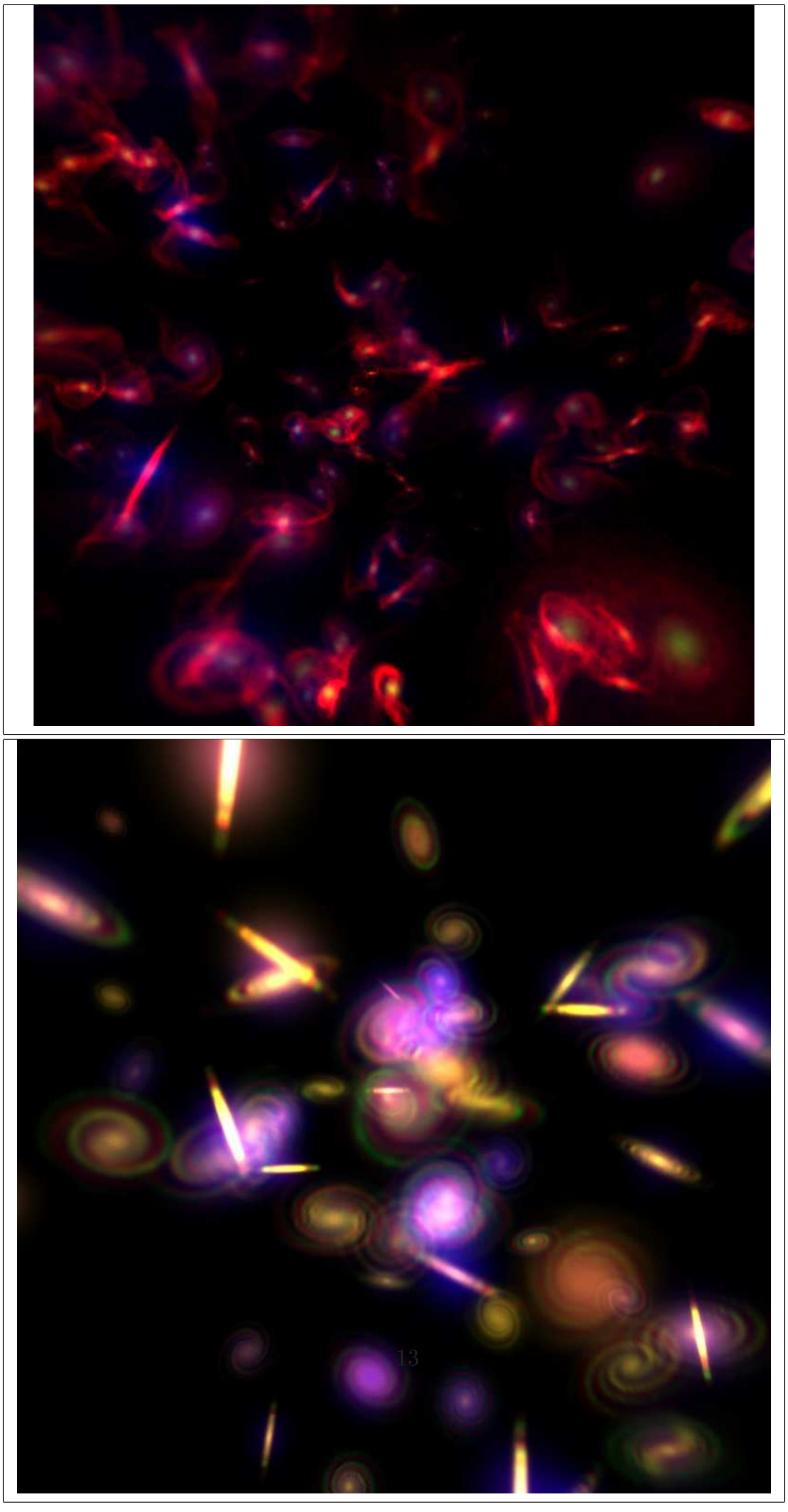


Figure 7: crowded spirals (top) and mergers (bottom) region

rescaleCat = scalar to rescale positions in cat;
if -1 will rescale according to dispersion in cat pos
threshDist= distance beyond which gals are converted into spheres
tmin= tmax= defines fraction of travelling to be rendered
extra= defines the extra objects to be added to the movie
EXAMPLE

```
gals=[]; for(i=1;i<=25;i++) grow,gals,setGal(is_random=1,gtype=1);
makeMovie("FirstMovie",gals,"input/line.trav",nsnap=20,res="sketch",run=0,
verb=4,size=350,threshDist=100,brightness=0.25,scale_trav=1.25);
```

- parseMpeg

duplicates and customize the inputfile for mpeg_encode;
possibly runs it;
EXAMPLE

```
parseMpeg("FirstMovie",range=[1,9]);
```

- readGal

reads ascii catalogue
rescale= scalar to rescale positions in cat; if 0 will rescale according to dispersion in cat pos
if negative will only rescale the size of objects
EXAMPLE

```
gals=readGal();
OR
gals=readGal("./input/grid2x2x2.cat",rescale=1.);
```

- readTrav

reads traveling file;

- setGal

sets galaxies either randomly
pos= position [x,y,z],
rot= rotation [x,y,z],
scale= scale float,
gtype= galactic type 1,2,3,4 1=spiral 2=elliptical 3=irreg 4= merger,
age= float (defines color of galaxy),
lum= float (defines luminosity of galaxy),
is_random= if set generates randomly the above,
seed= if set generates randomly the above using seed,
df3= string containing the df3 cube to be used; if void will be set according to gtype
EXAMPLE

```
gals=[]; for(i=1;i<=5;i++) grow,gals,setGal(is_random=1);
//tt=df3_read(gals(1).df3+".r"); pli,tt(sum,,);
```

- writeCat

exports galaxies into fname
EXAMPLE

```
writeCat("input/crap.cat",gals);
```

- writeGal

```
Parse galaxies into povray stream
   FLAGS
   brightness = 1; // larger brighter
   scale = 1.; // distance to objects
   viewpoint=[1,0,0] // position of camera
   viewdir=[0,0,0] // orientation of camera
   viewangle= 60 (in degree) // view angle of camera
   Qnbray=1; // nb rays used (should be >=1 ) larger slower
   Qconf= // confidence level for ray tracing
   Qnbbloc=1; // nb blocks used (can be <1 ) larger slower
   Qvar= 1/100; // variance for ray tracing
   EXAMPLE
   ff=open("test.pov","w"); writeGal(ff,gals(1:5)); close,ff;
```

- `writeTrav`

```
writes traveling file
```

B Description of the hyper icons

This section is bound to become rapidly obsolete since this implementation is rather crude. All hyper icons (except for the irregulars) are cubes constructed from analytical profiles.

B.1 Descriptions

- Ellipticals are elliptical 3D cirsic profiles.
- Spirals are logarithmic spirals with exponential bulge (in z) and Gaussian in x, y. The level of winding is varied together with the size of the bulge and the number of arms.
- Bar spirals are like spirals except that the inner region does not wind; it always has 2 arms and the departure from winding is controled by some powerlaw.
- Irregulars are Gaussian random fields required to vanish beyond an elliptical region.
- Mergers are extracted from collisions of galaxies carried in Gadget with Magalie as initial conditions.

B.2 Colour mixture

Again the process is rather crude; a elliptical is a (r, g, b) mixture of 3 cubes; a spiral and a barred spiral is a mixture of a blue spiral (resp. barred spiral) and a red and green S0; the age of the galaxy is coded so that the blue component decreases with age.

B.3 Possible improvements

For the spirals, add external spirals as well as the central one, together with ofset spirals which absorb. Define base colour for the 3 cubes which are not (r, g, b) but rather blueish, redish, yellowish so that the 3 components can be varied independantly of colour.

C What's new ?

- more realistic multicomponents spirals
- mergers
- variable field of view and opening angle