Cloudy in 3D

Christophe Morisset
Instituto de Astronomía, UNAM
Mexico

Summary

- Why 3D
  - Images : fitting morphologies
  - Aperture effects
  - Velocity, line profiles, PV diagrams
- How 3D, short history
  - São Paulo, MOCASSIN, Cloudy_3D (IDL, Python)
  - Limitations & speed
 - 3D reconstruction (Tomography, SHAPE)
- Examples of use

Why 3D ?

- As we all know, PN are spherical, and HII regions plan parallels :-)

# Not O nor //

NGC 2346

NGC 6543   HST · WFPC2
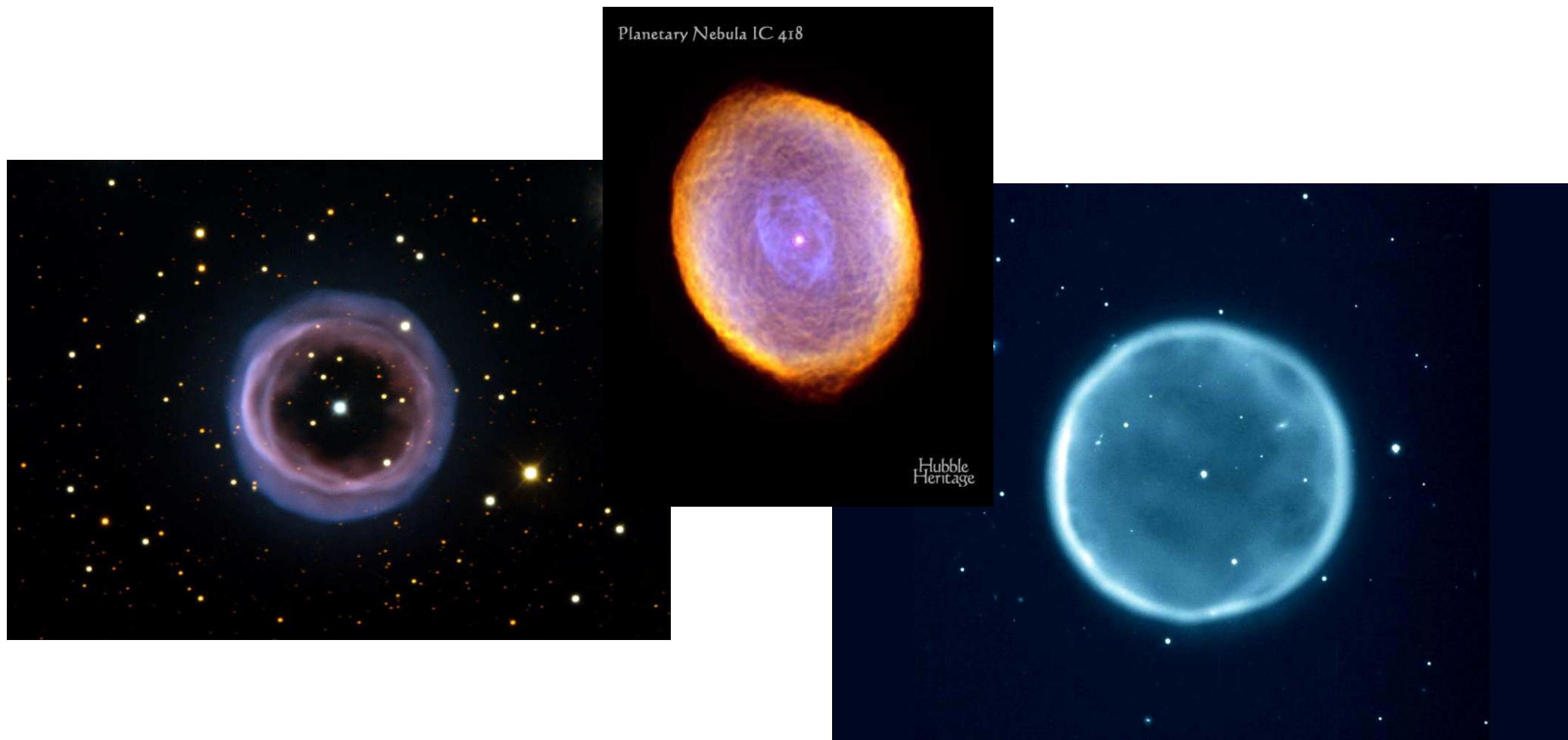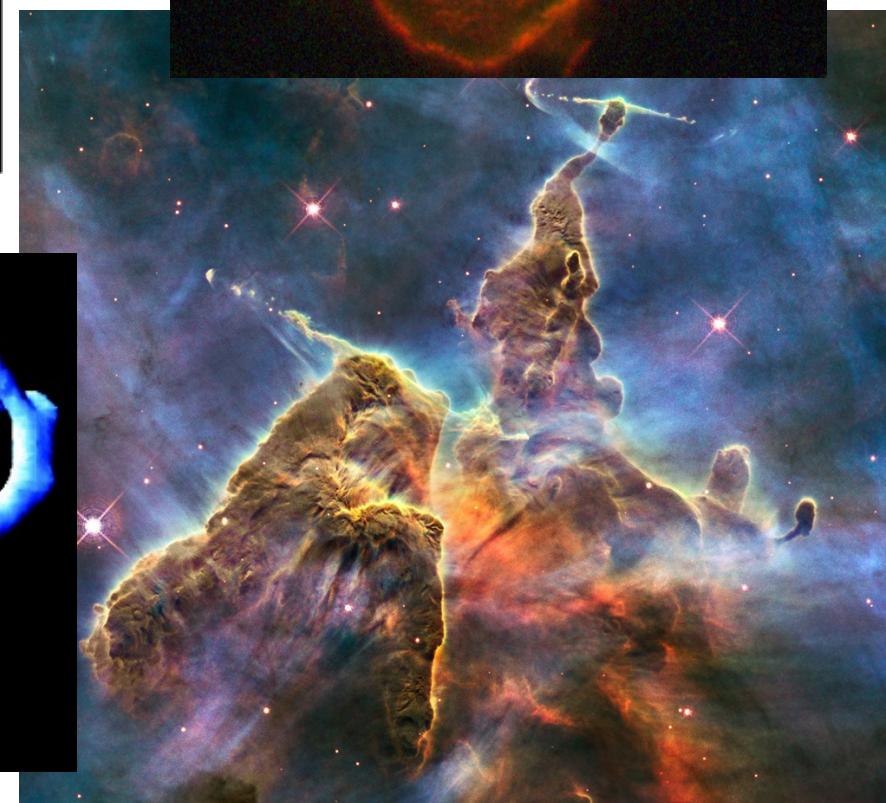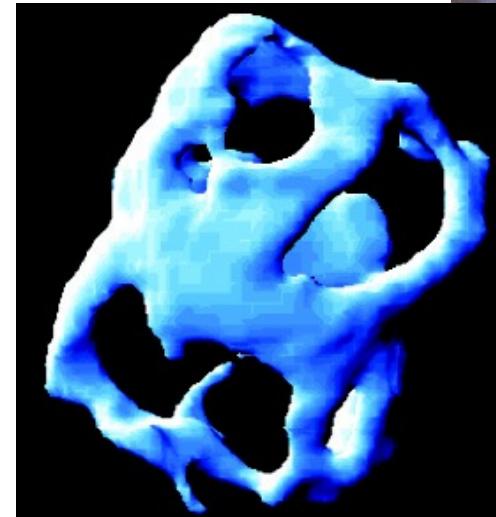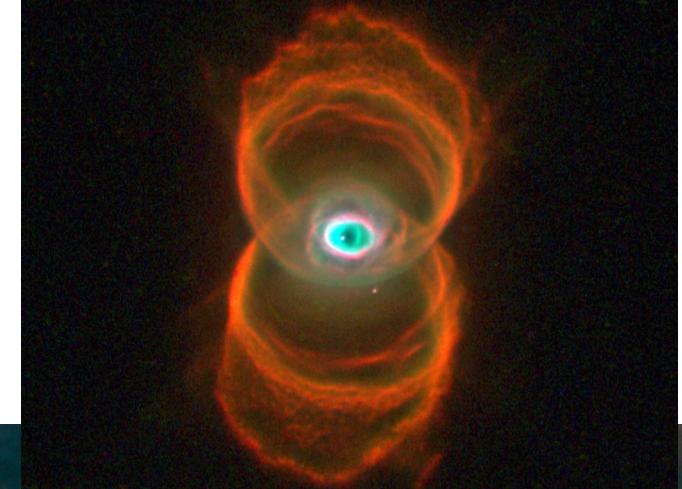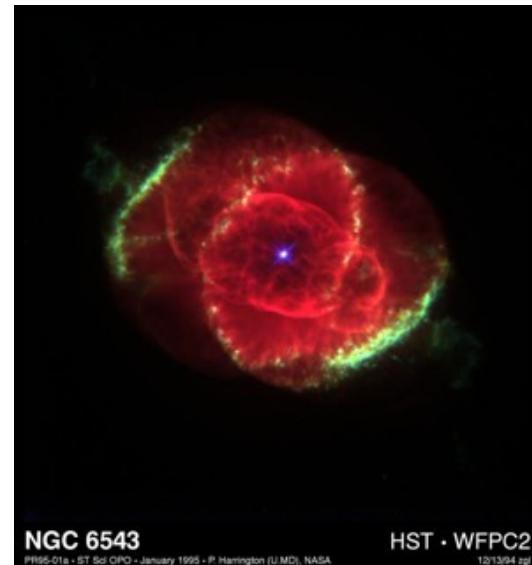
Aperture effects
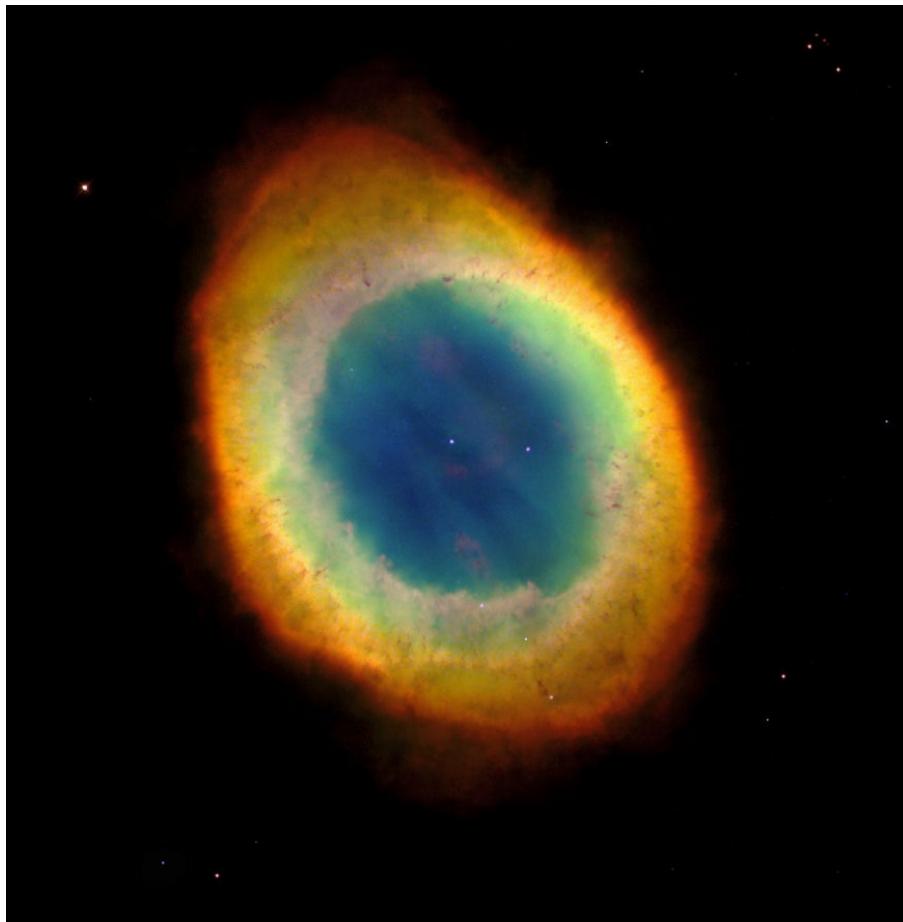
- Color = dominating emission line.

- Position : different line ratios.

Velocity field

- Velocity field : line-of-sight component projection.
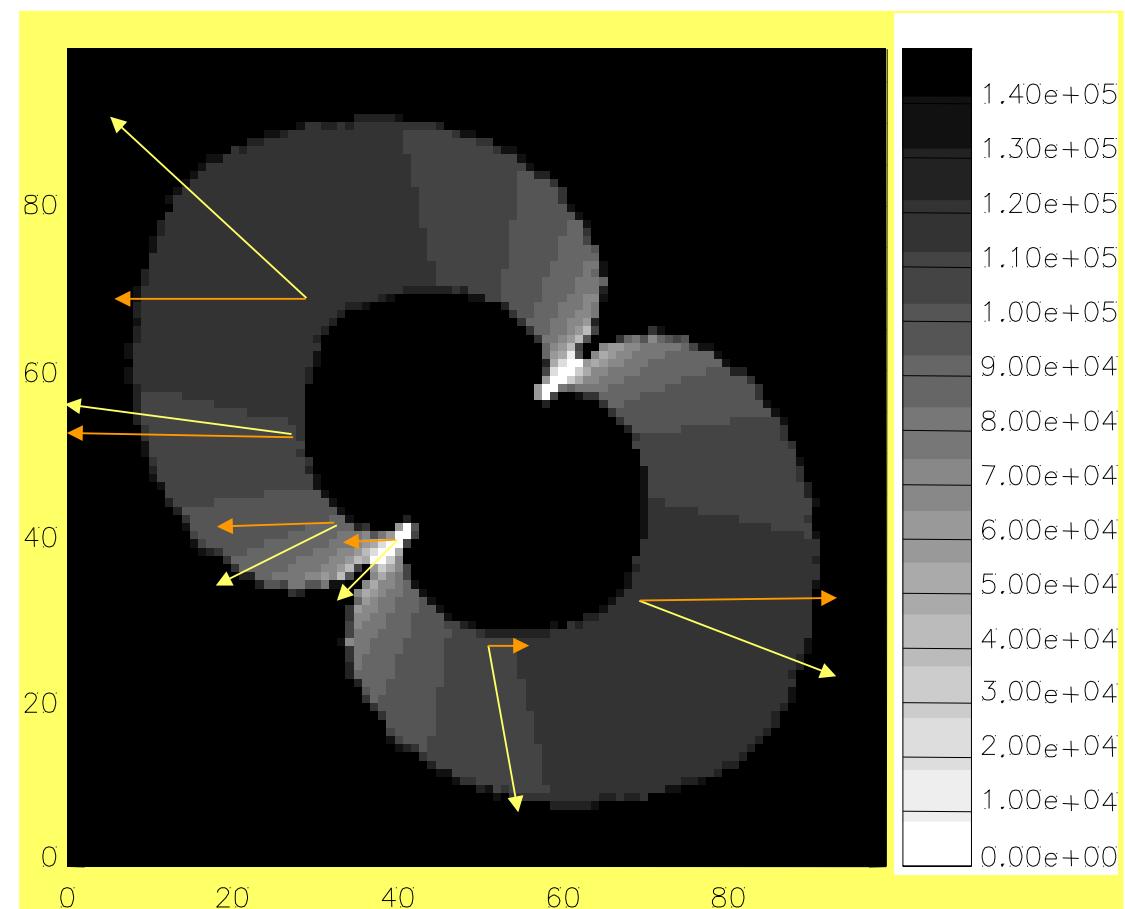
MODELLING OF ASYMMETRIC NEBULAE. II. LINE PROFILES

C. Morisset[1] and G. Stasińska[2]
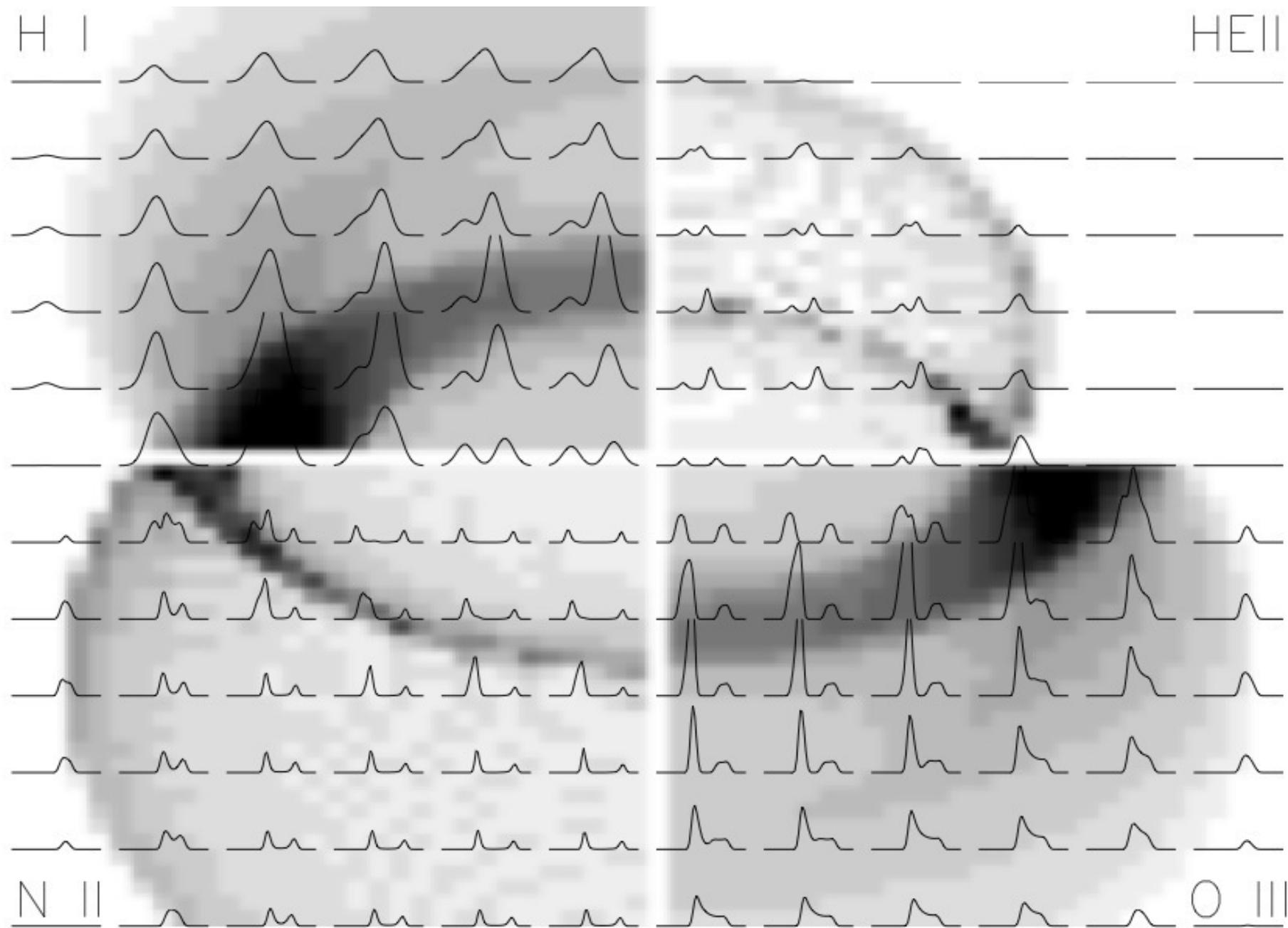
PV-diagrams
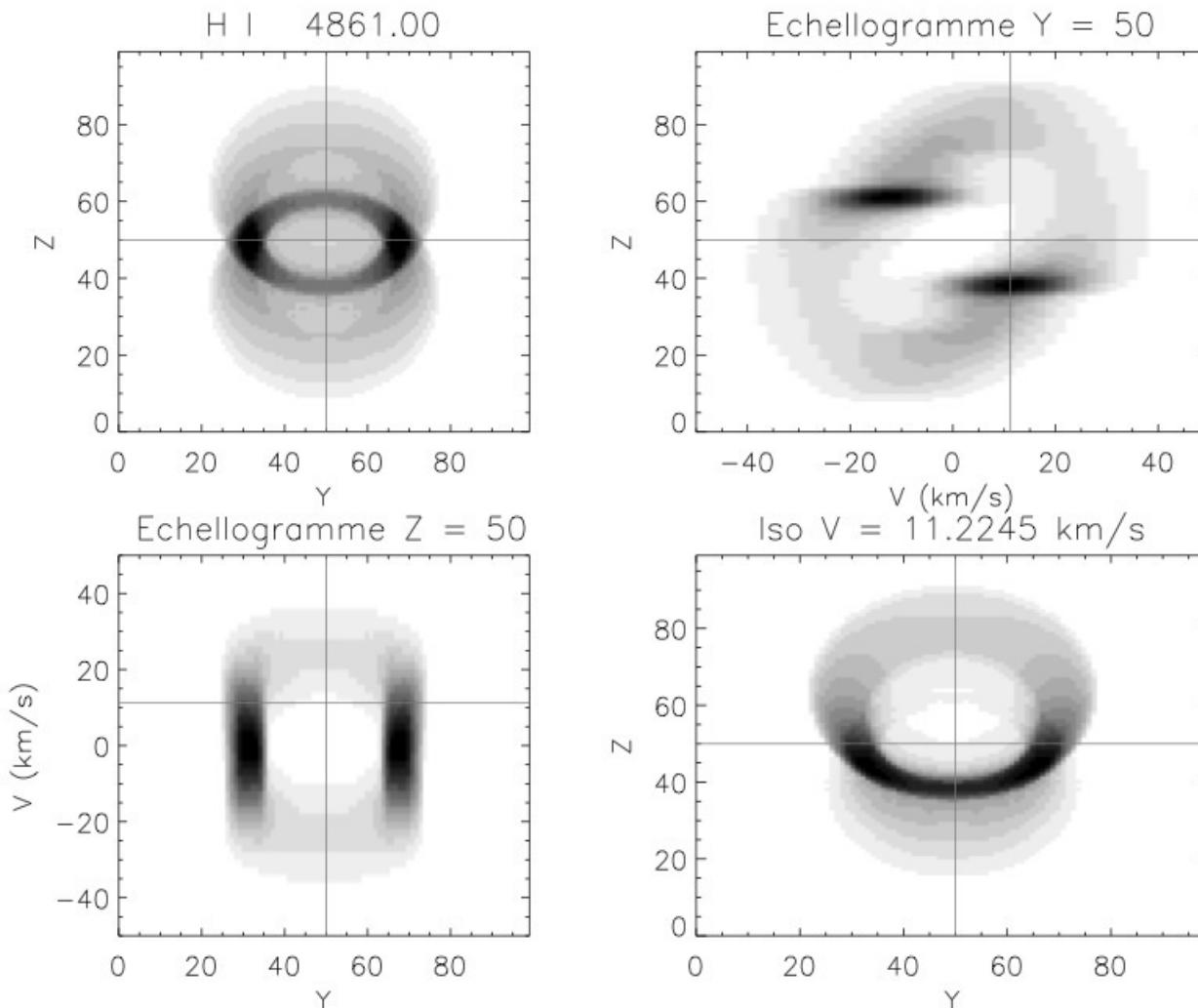
Fig. 10. PV diagrams obtained for the bipolar nebula presented in § 3.1. Upper left: surface brightness image for Hβ with the 2 slits used for the forthcoming PV diagrams, Upper right: PV diagram obtained for a narrow vertical slit crossing the center of the nebula, Lower left: PV diagram obtained for a narrow horizontal slit crossing the center of the nebula, Lower right: velocity channel map obtained from the V=11 km s^{-1} channel.
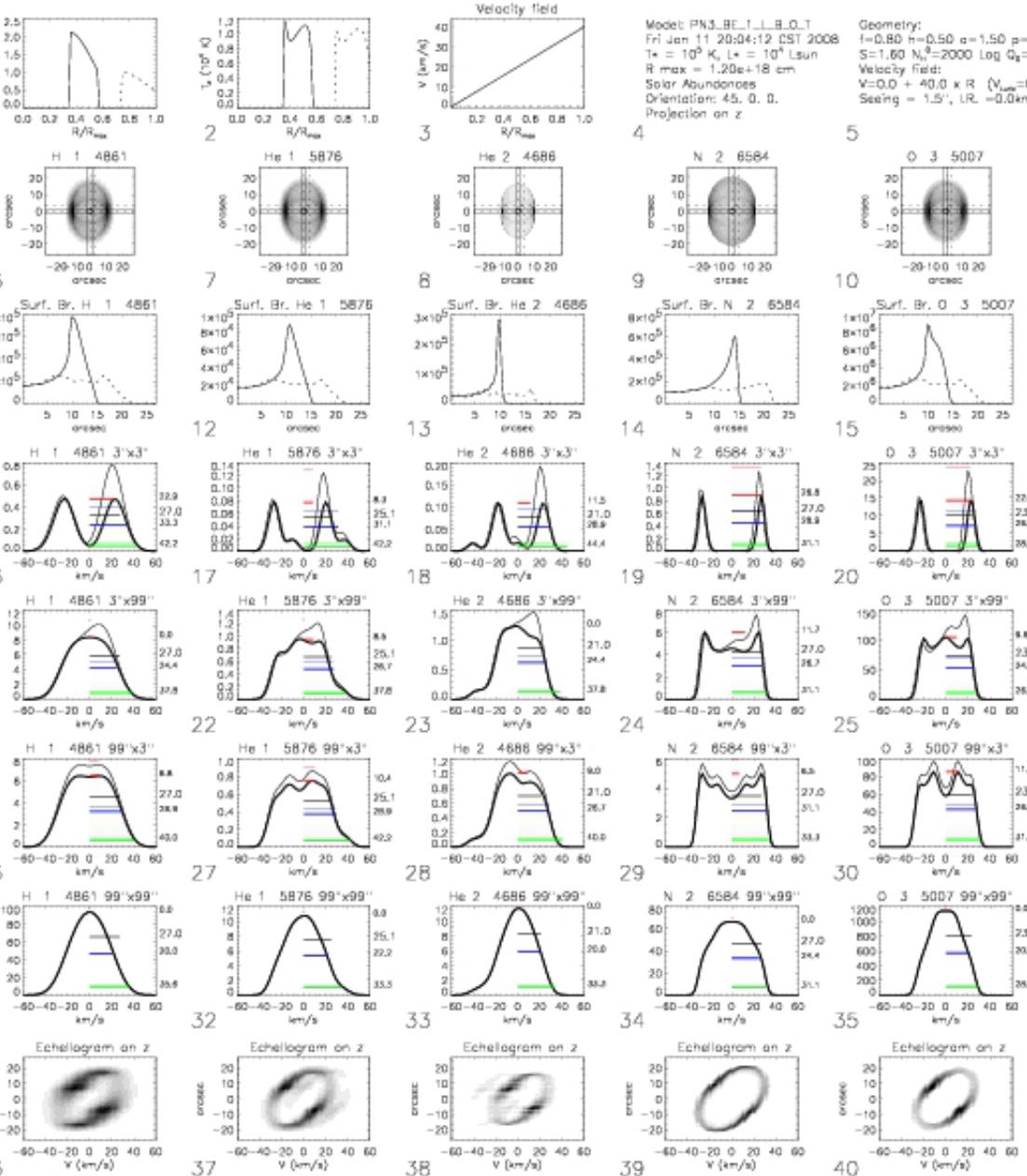
AN ATLAS OF SYNTHETIC LINE PROFILES OF PLANETARY NEBULAE

C. Morisset[1] and G. Stasińska[2]
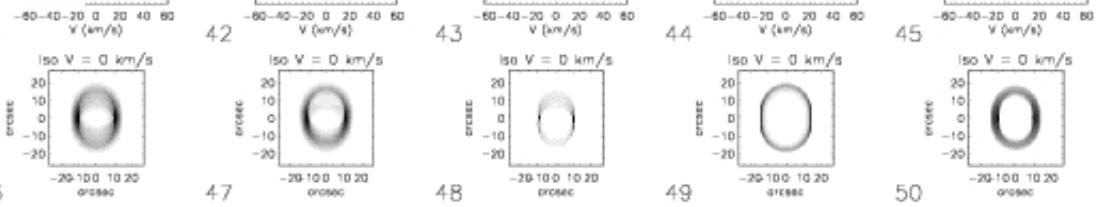
3D photoionization codes
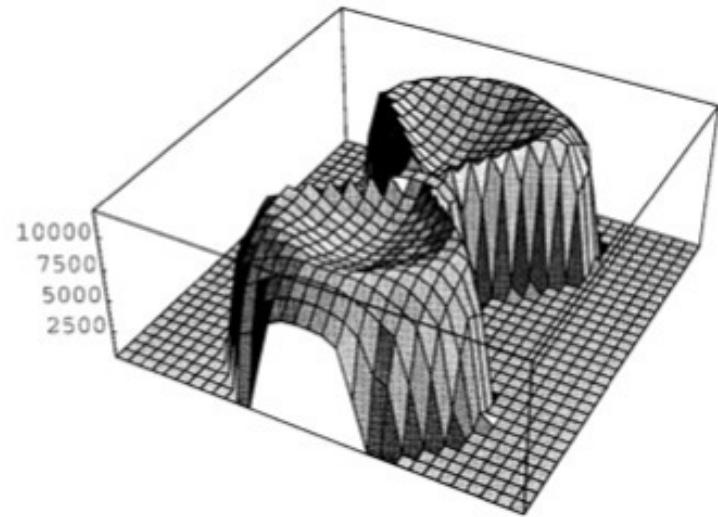
Fig. 1.—Electron temperature in the plane (110″ × 110″) perpendicular to the torus. The vertical axis gives the temperature (K).

A NEW GENERATION OF PHOTOIONIZATION CODES: THREE-DIMENSIONAL MODELS. THE BIPOLAR PLANETARY NEBULA IC 4406

R. GRUENWALD AND S. M. VIEGAS

Instituto Astronômico e Geofísico/Universidade de São Paulo, Av. Miguel Stefano, 4200, 04301-904, São Paulo, SP, Brazil

AND

D. BROGUIÈRE
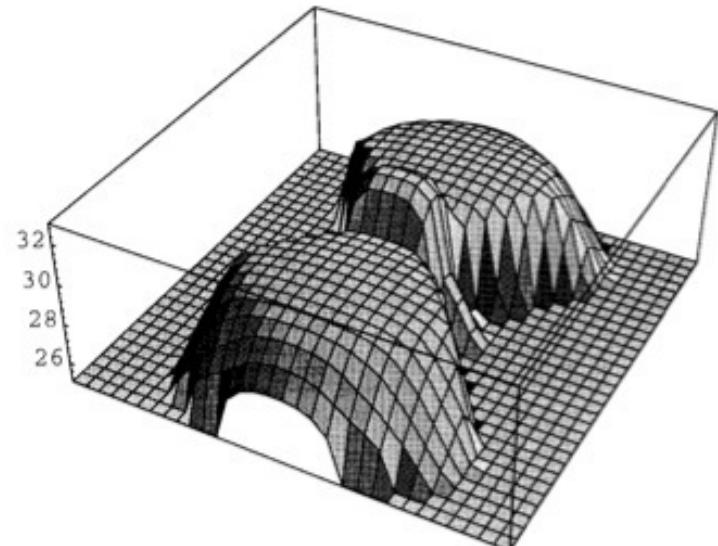
Fig. 2.—Intensity distribution of Hα (arbitrary units) in a 110″ × 110″ surface projected in the plane of the sky.

MORPHOLOGY AND KINEMATICS OF PLANETARY NEBULAE. I. A NEW MODELING TOOL

C. Morisset,[1,2] R. Gruenwald,[2] and S. M. Viegas[2]
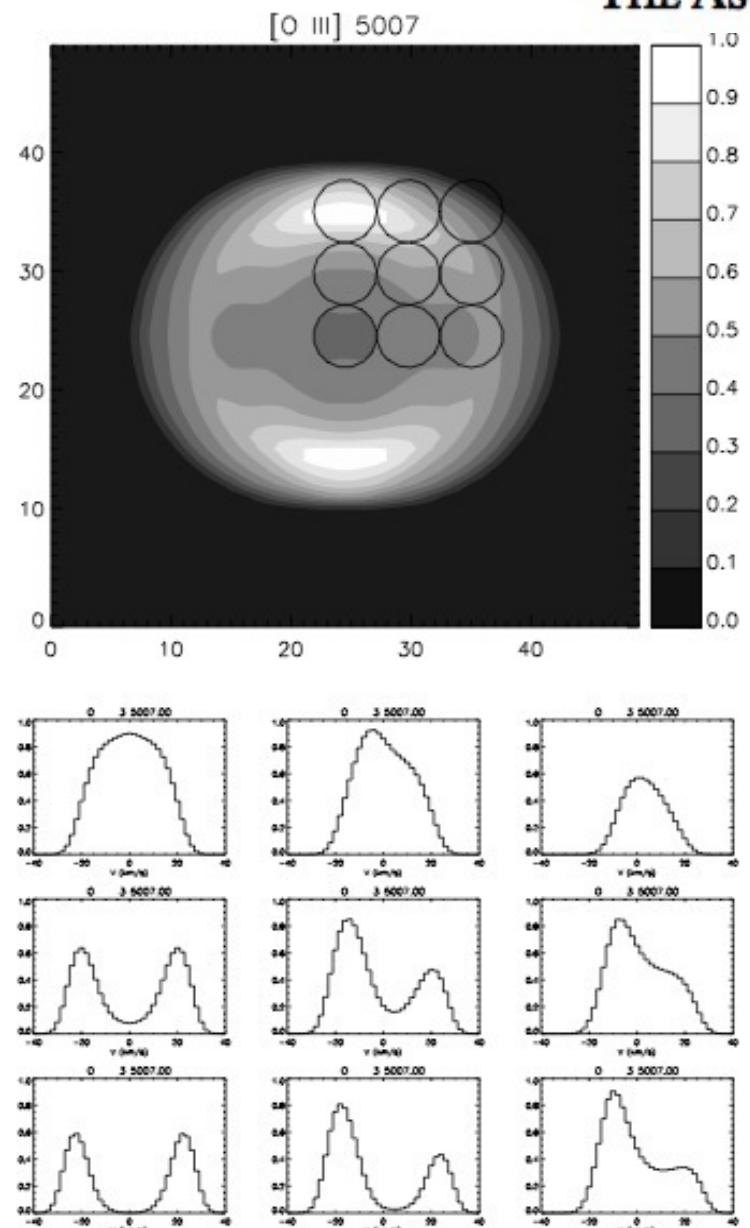
Fig. 4.—Top: Size and positions of the aperture used to obtain the [O III] velocity profiles shown in the bottom nine panels for model A1. The nebula is rotated by 50° around the z-axis and projected on the y-direction. The axes are in pixels. Bottom: The emission-line profiles are normalized.
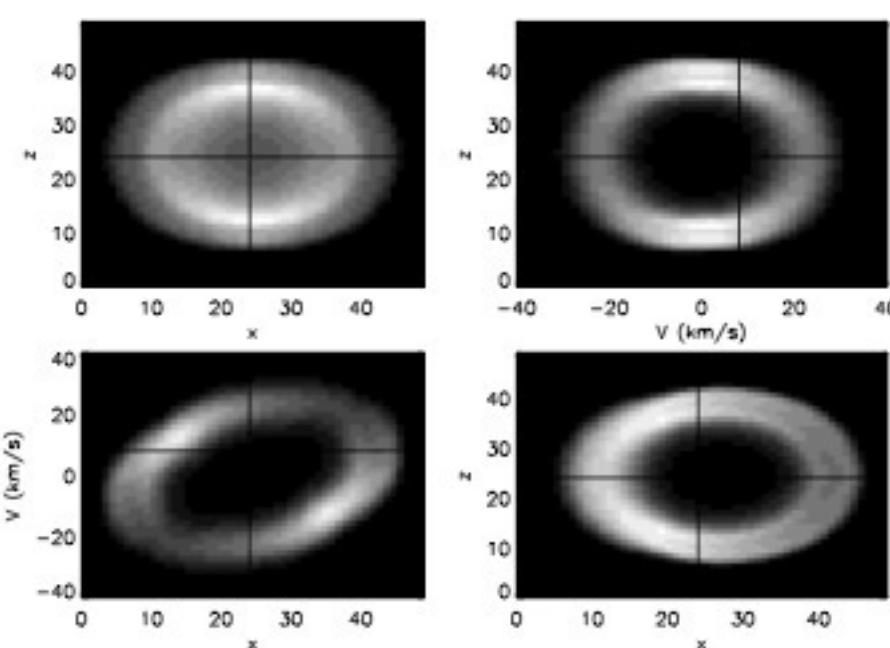
Fig. 5.—[O III] line image (top left), P-V diagrams obtained through centered horizontal (top right) and vertical (bottom left) slits, and iso-velocity contours for $V = 9$ km s^{-1} (bottom right). The z- and y-axes are given in pixels.

Fig. 4.—Gas distribution for the Diabolo model. Only the denser zone (1300 cm^{-3}) is shown.

images from the Diabolo model. From the upper left to the lower right panel, the angle between the axis of symmetry and the sky plane ' from 0° to 80°.

Och's Monte-Carlo code

# Diffuse radiation in models of photoionized nebulae

S.R. Och[1], L.B. Lucy[2], and M.R. Rosa[2],*

[1] Dr. Remeis-Sternwarte, Astronomisches Institut der Universität Erlangen-Nürnberg, Sternwartstrasse 7, D-96049 Bamberg, Germany,
[2] Space Telescope-European Coordinating Facility, European Southern Observatory, Karl-Schwarzschild-Strasse 2,
  D-85748 Garching, Germany (llucy@eso.org)
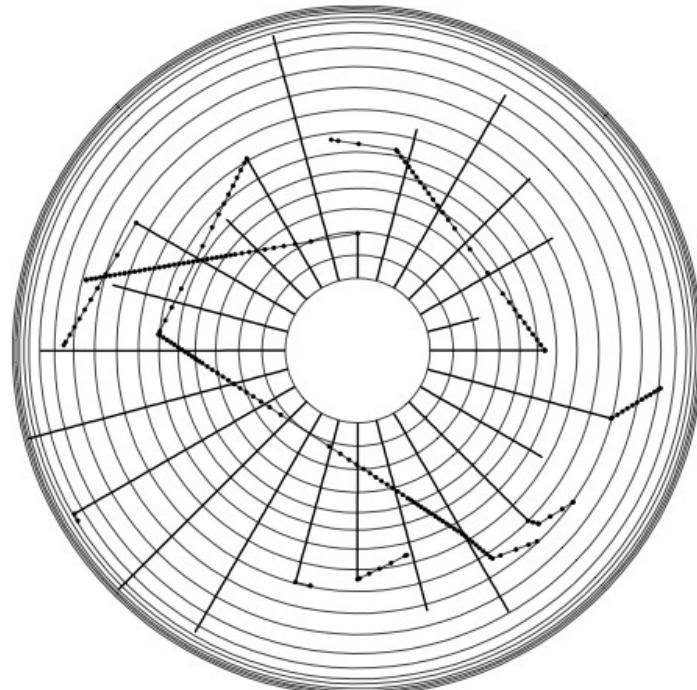
Fig. 1. The random walks for several photon packages starting at the inner nebular radius are plotted. Stellar photon packages are shown as thick lines, while diffuse packages passing reference positions are represented by filled dots (only every fifth radial grid surface is drawn). Once they have turned into non-ionizing packages, they are no longer tracked since they escape immediately.

MOCASSIN: a fully three-dimensional Monte Carlo photoionization code

B. Ercolano,[*] M. J. Barlow, P. J. Storey and X.-W. Liu[†]

University College London, Gower Street, London WC1E 6BT

Three-dimensional photoionization modelling of the planetary nebula NGC 3918

B. Ercolano,[1][*] C. Morisset,[2][†] M. J. Barlow,[1] P. J. Storey[1] and X.-W. Liu[1][‡]

[1] *University College London, Gower Street, London WC1E 6BT*
[2] *Laboratoire d'Astrophysique de Marseille, CNRS, BP 8, 13376 Marseille Cedex 12, France*
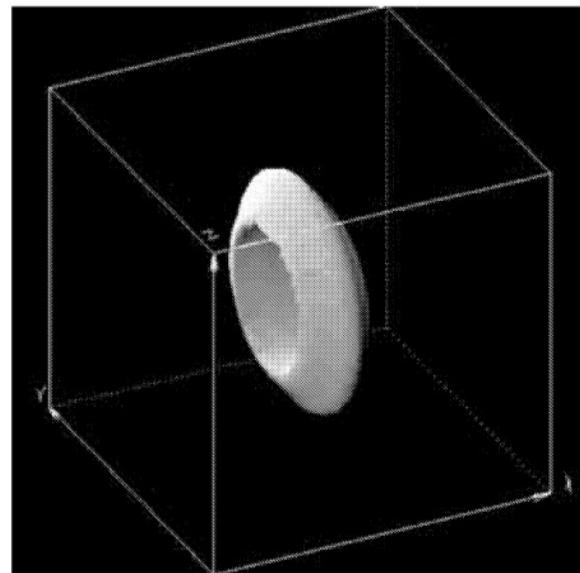
Figure 3. Three-dimensional isodensity plot of the dense torus component of spindle-like model A.
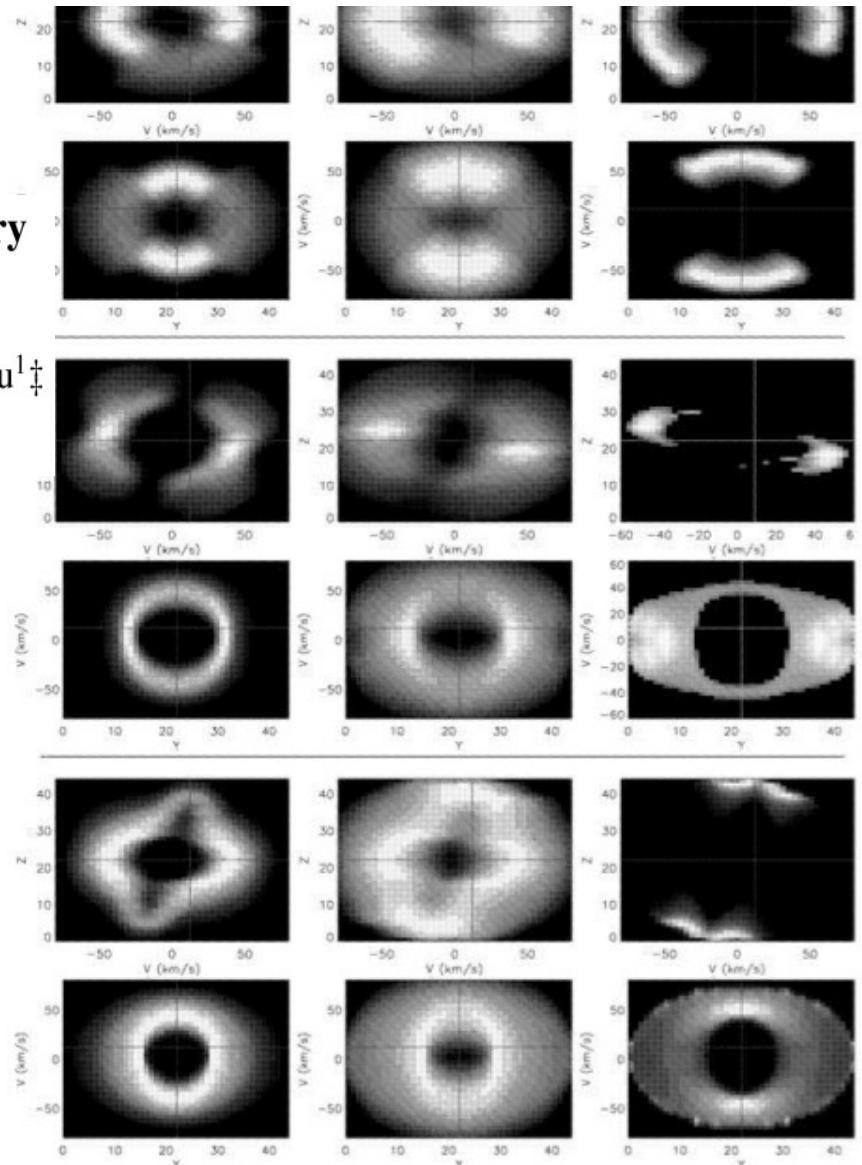
Figure 11. Position–velocity diagrams in He II 4686Å (left panels), Hα (central panels) and [N II] 6584Å (right panels) for the biconical model (top two rows), spindle-like model A (middle two rows) and spindle-like model B (bottom two rows) of NGC 3918. The top panels in each pair of rows show the echellograms obtained with slits oriented at PA = −76° while the bottom diagrams show echellograms obtained in the same lines, but with slits positioned at PA = +13°.

MOCASSIN and Cloudy_3D

- MOCASSIN : A full 3D photoionization code, using Monte-Carlo.

- F90 MPI code, running on cluster.

- Few hours to days to run/converge a model (cluster allocation time policies).

- → need for a quick (but not so dirty) code to obtain « pseudo-3D » models

- Cloudy_3D : we're loosing the « full » 3D, but a few minutes to run:-)

Cloudy_3D

C. Morisset,[1]★ G. Stasińska[2] and M. Peña[1]
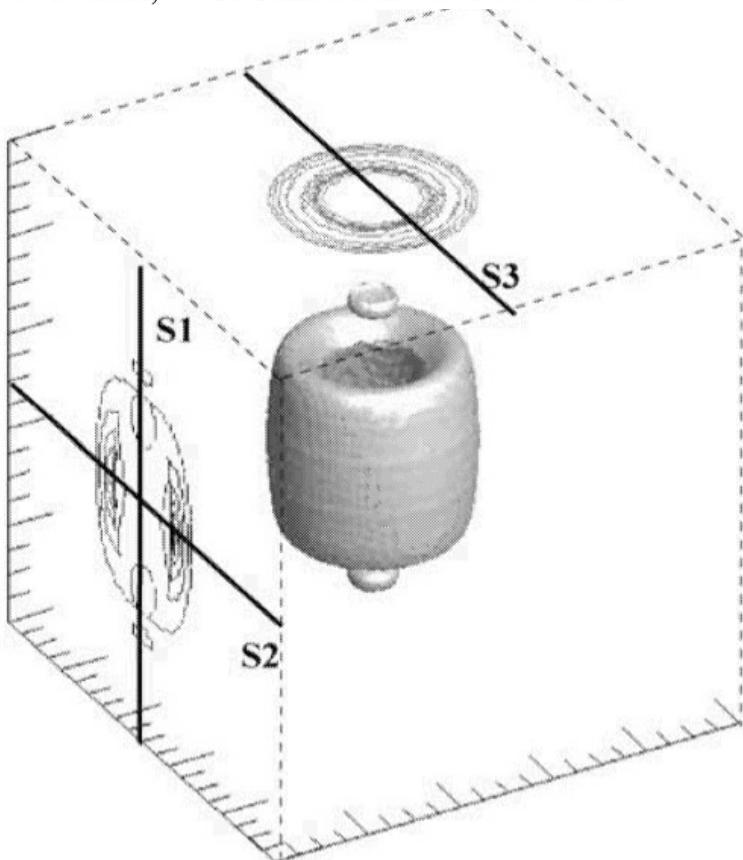
Figure 2. 3D representation of the nebula. An isodensity surface is drawn, showing the equatorial density enhancement and the two polar knots. On the faces we have represented the Hβ surface brightness contours for two orientations of the nebula: on the left side, for the nebula seen with the polar axis parallel to the sky; on the top, for the nebula seen pole-on. The slits used to determine parameters from emission-line ratios are indicated.

- Various runs of Cloudy (1D), corresponding to different angles.

- 3D reconstruction in a coordinate cube by interpolation between the 1D runs.
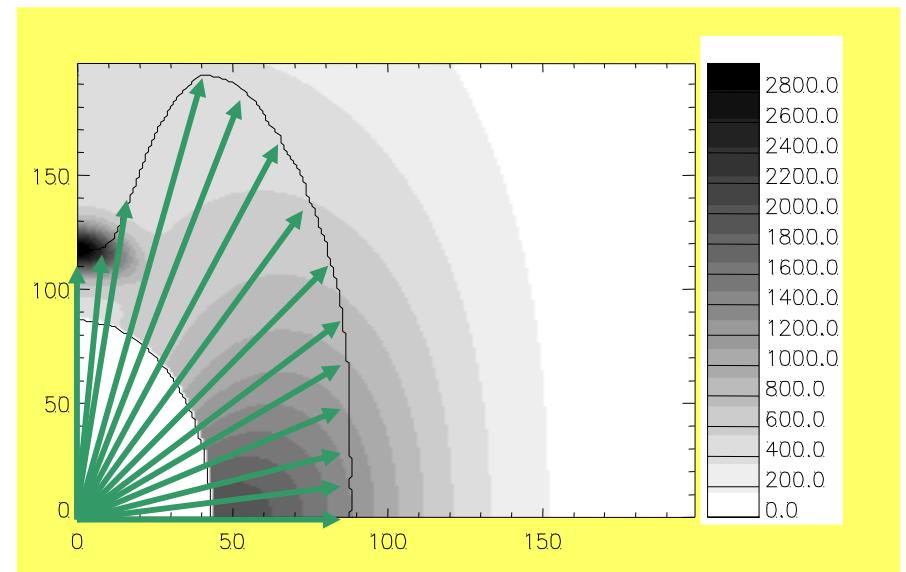
C3D :faster but limited ?

•Cloudy_3D is not a « full » 3D code. It only considers radial radiation (as a combination of 1D runs).
•Limitation : when no-radial radiation dominates a process (e.g. Photoionization).
•2 cases :
•Shadows
•Multiple stars
Otherwise : C3D is OK !

Lights in the Shadows, 3D-Modelling Knots with MOCASSIN

Christophe Morisset

Instituto de Astronomia, UNAM, Mexico
<morisset@astroscu.unam.mx>

Barbara Ercolano

University College of London, UK
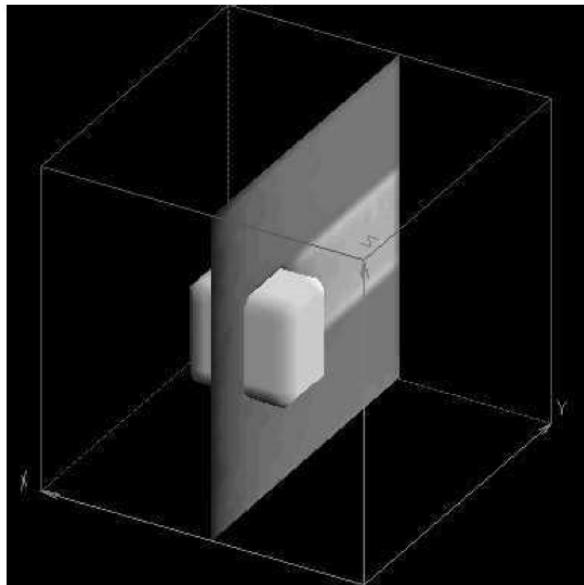
Figure 1.   Density distribution of the gas for the model of a Knot. The plane parall ionizing flux is arriving from bottom-left.

HeII   [NII]   [OIII]

[SII]   [NeIII]   [NeIV]
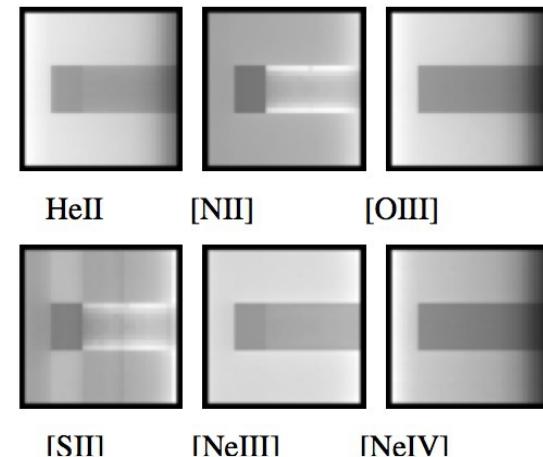
Figure 3.   Synthetic images in the sky plane, for some emission lines. The main trends observed in the Helix Knots are reproduced : recombination and forbidden lines from low charged ions are emitted preferentially by the tail.

The effects of spatially distributed ionisation sources on the temperature structure of H II regions

Barbara Ercolano · Nate Bastian · Grazyna Stasińska
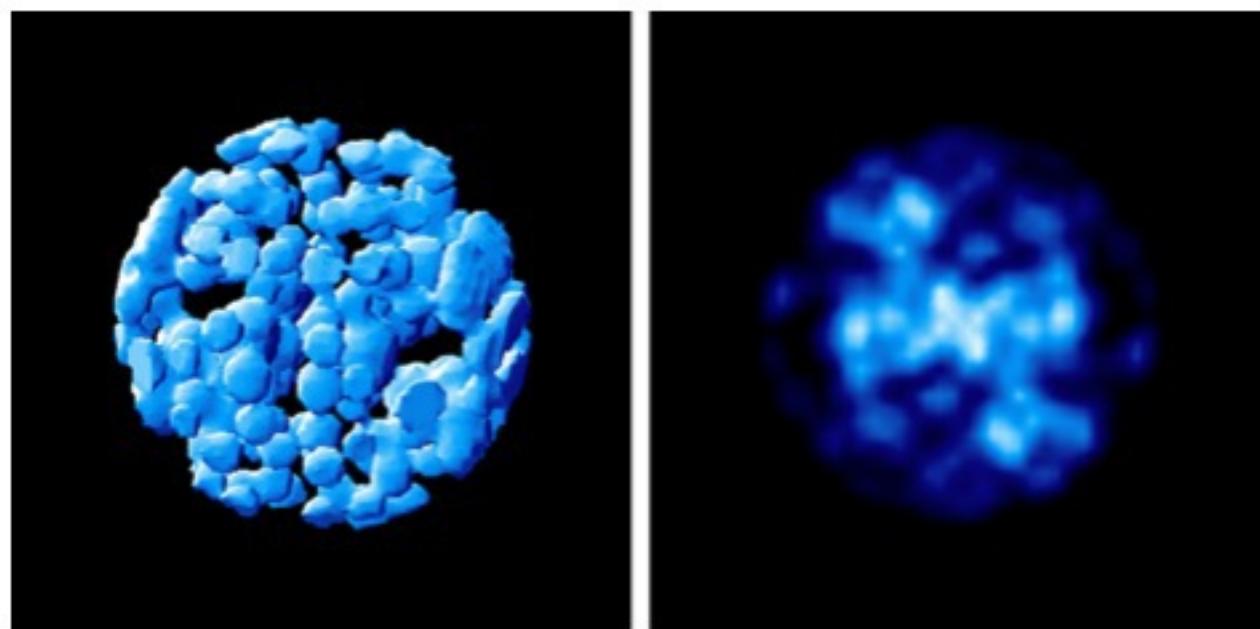
Fig. 1 *The left-hand panel* shows a 3D representation of the Strömgren sphere distribution for case F, plotted as the iso-surfaces where the ionisation fraction of hydrogen is 0.95. *The right-hand panel* shows an average projection map of the ionic abundance of H^+

3D reconstruction

THE THREE-DIMENSIONAL IONIZATION STRUCTURE AND EVOLUTION OF NGC 6720, THE RING NEBULA[1,2]

C. R. O'DELL

Department of Physics and Astronomy, Vanderbilt University, Box 1807-B, Nashville, TN 37235, USA; cr.odell@vanderbilt.edu

F. SABBADIN

INAF-Astronomical Observatory of Padua, Vicolo dell'Osservatorio 5, 35122 Padua, Italy

AND

W. J. HENNEY

Centro de Radioastronomía y Astrofísica, UNAM Campus Morelia, A.P. 3-72, 58090 Morelia, Michaoacán, Mexico
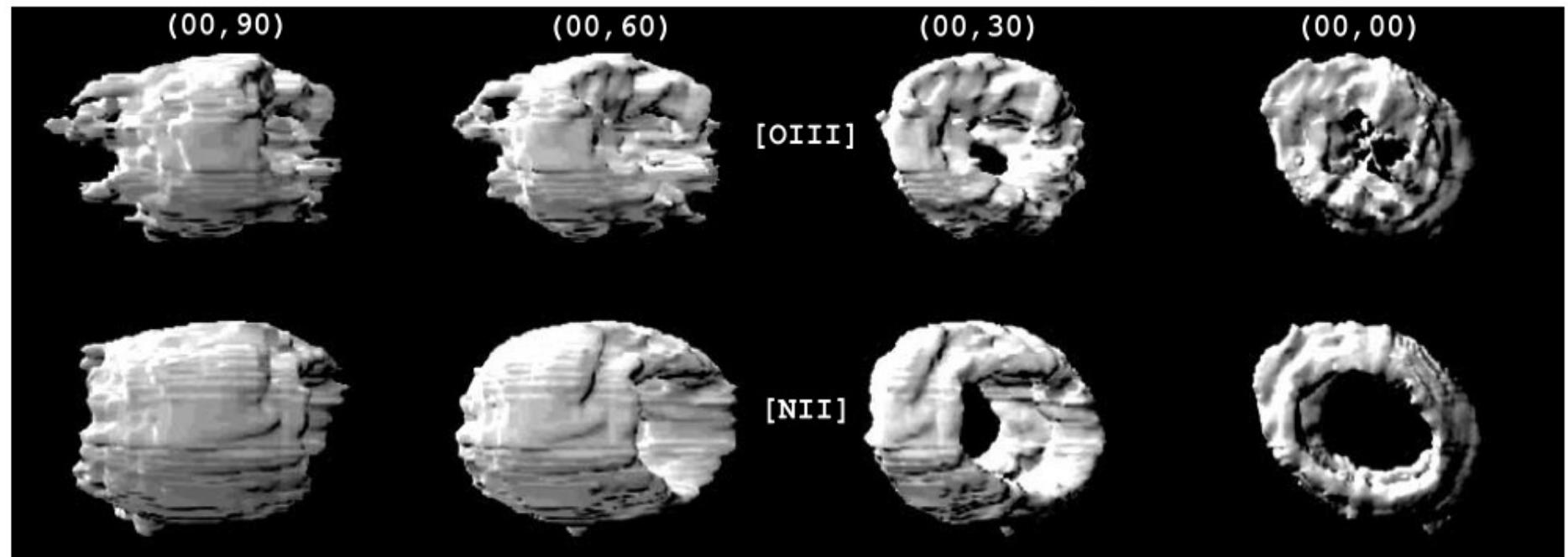
FIG. 8.—Opaque reconstruction of NGC 6720 in [O III] and [N II] at the mean flux levels. Same orientation as Fig. 7.

3D reconstruction
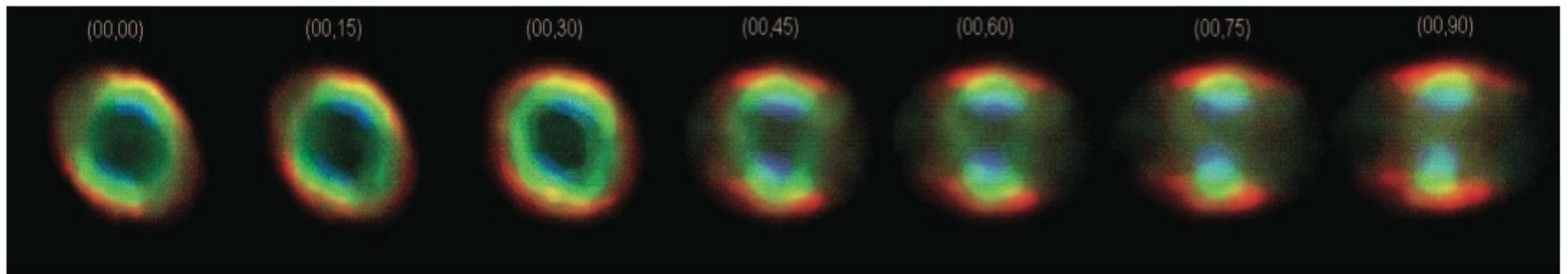
FIG. 10.—Optical appearance of NGC 6720 (He II, *blue*; [O III], *green*; [N II], *red*) for a rotation through the north-south axis centered on the exciting star. The leftmost panel, (00, 00), corresponds to the rebuilt nebula as seen from the Earth (north is up, and east to the left). [*This figure is available as part of an mpeg animation with Fig. 7 in the electronic edition of the Journal.*]

Shape: A 3D Modeling Tool for Astrophysics

Wolfgang Steffen, Nicholas Koning, Stephan Wenger, Christophe Morisset, and
Marcus Magnor, *Member, IEEE*
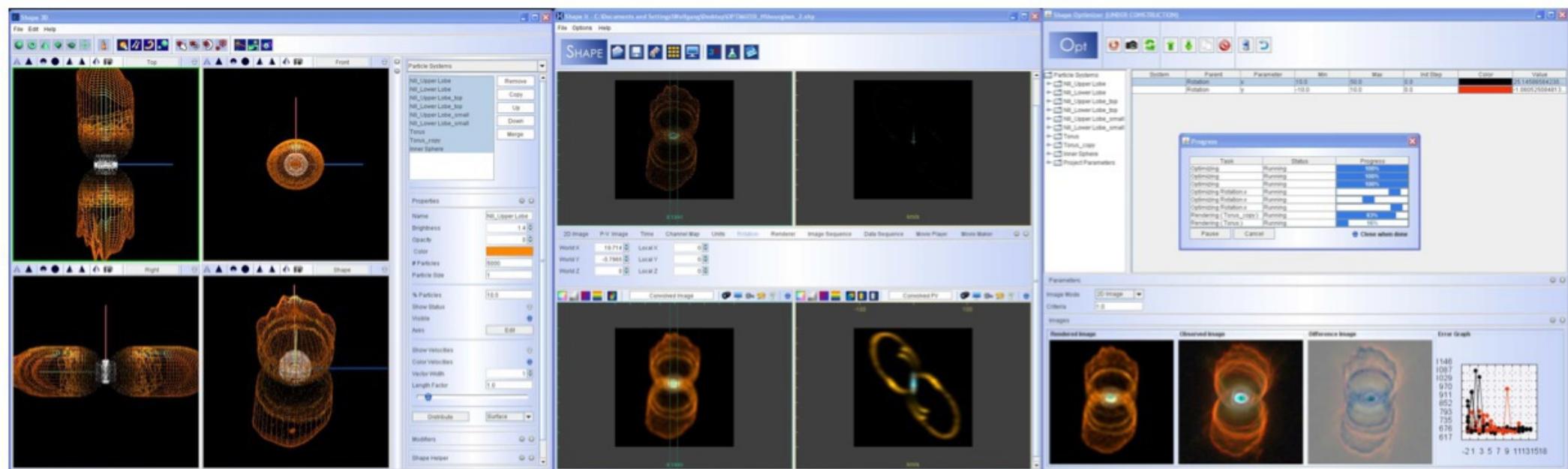
Fig. 4: The *Shape* graphical user interface contains several modules dedicated to different tasks. Here, the interactive 3D modeling window, the screen for rendering and comparison to observational data and the control panel for automatic optimization are shown, respectively, from left to right.

Shape
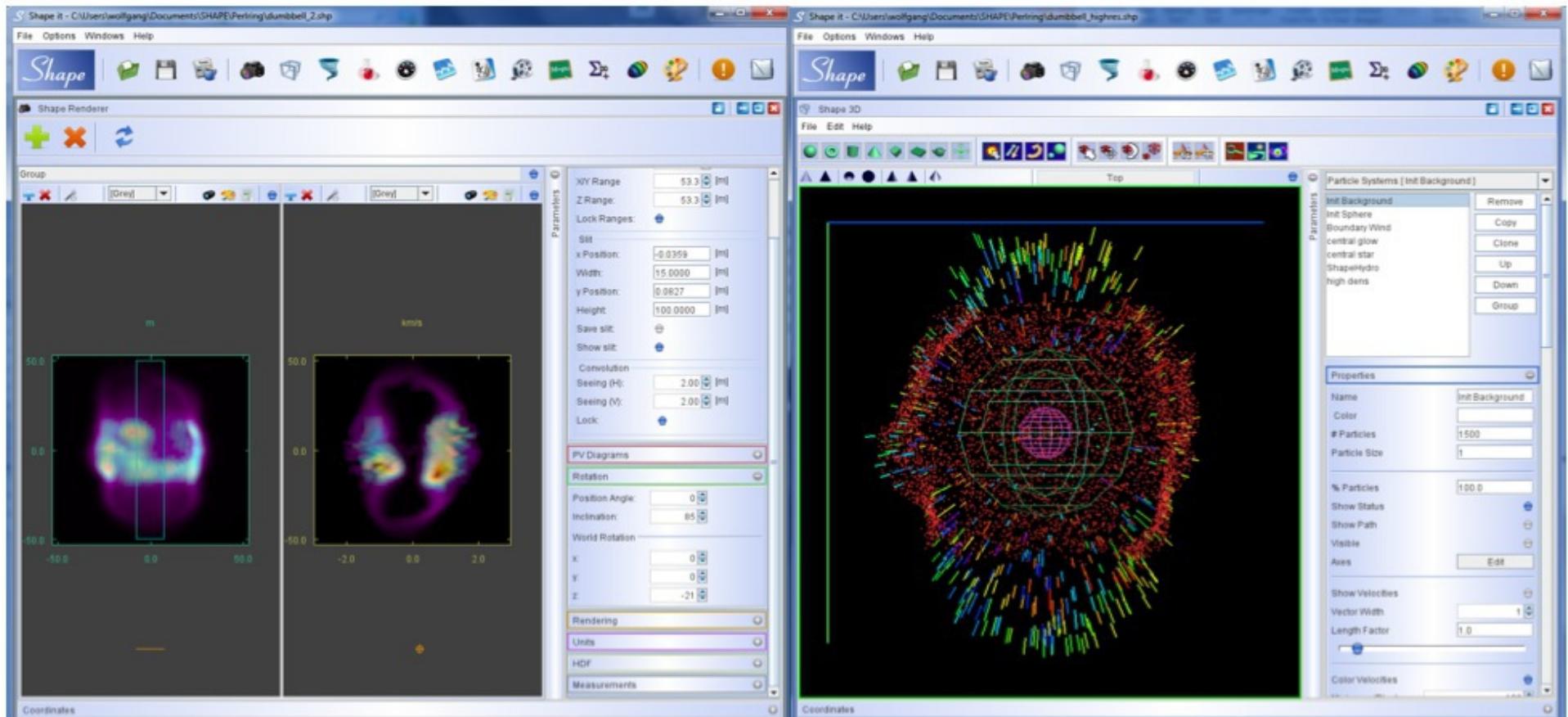
Fig. 4. Screenshots of the rendering (left) and 3D modeling (right) modules of the *Shape* software. The portrayed nebula was generated using *Shape*'s hydrodynamical simulation capabilities. The rendering module shows a false-color image (left) and a visualization of the Doppler shift of the spectral lines (right). The screenshot of the 3D module illustrates the structure and velocity information obtained from the simulation.

Shape

LÓPEZ ET AL.
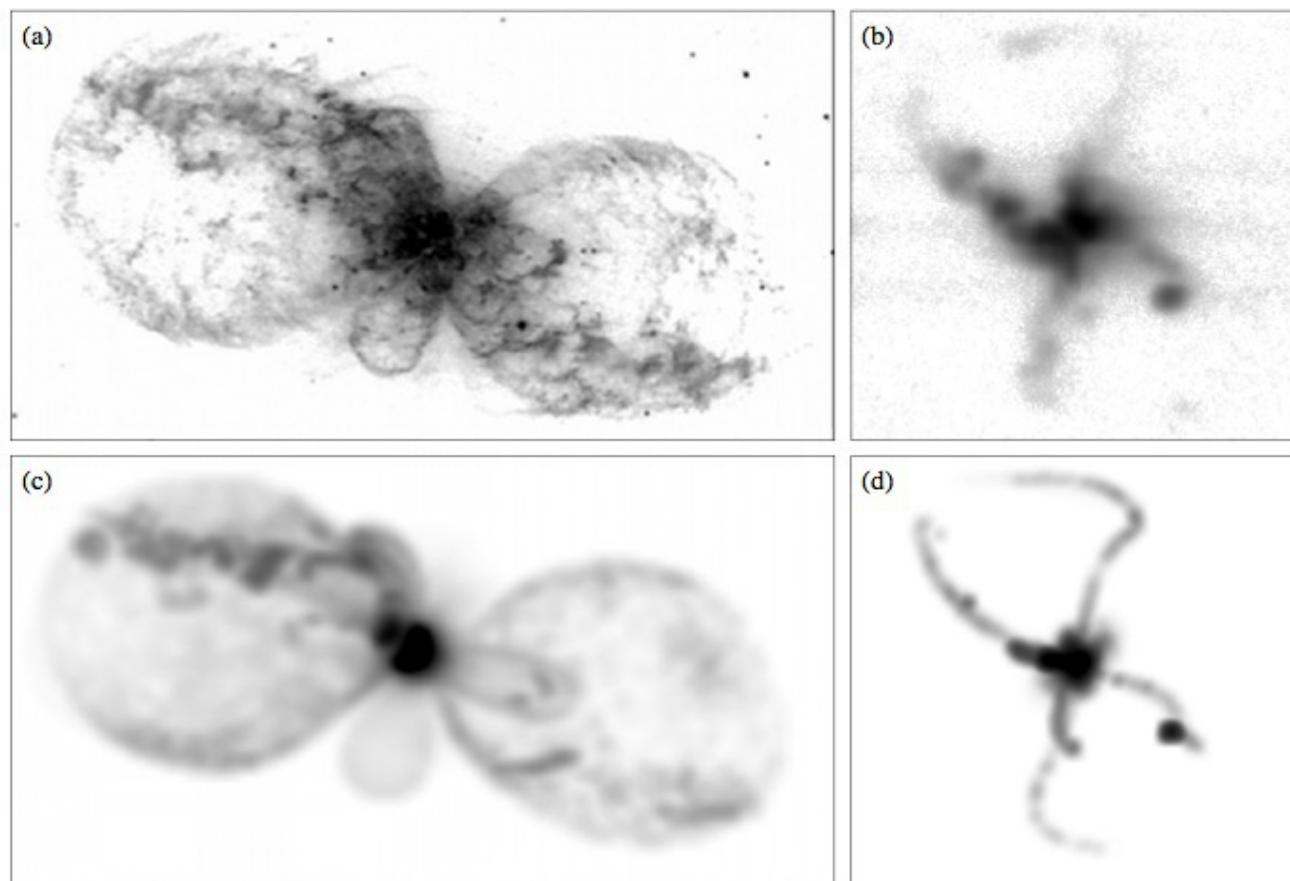
Figure 8. Top panels, frames (a) and (b), show the *HST* image of Hb 5 and the observed [N II] *P–V* array from slit n. Lower panels, (c) and (d), are the synthetic image and synthetic [N II] *P–V* array from slit n, modeled with SHAPE.

Using C3D and pyCloudy

- C3D is actually a library running and dealing with the results of Cloudy.

- First implementation using IDL.

- The new version is in **Python**.

- User manual not yet available, sorry... In a few months I hope :-)

- Object-oriented.

- Can also be used for a 1D run, or for grids.

Run a model

- A quite simple object is used to define and write the Cloudy input file. It allows to automatically add the various « save » options, with the predefined extension.

Read a model

- An object exists that reads the outputs of a Cloudy run and fill variables.

- Methods are available to compute some integration on the radius or volume (weighted by different parameters).

- T0(line), T0(ion), t2(line), t2(ion) are computed.

- Model can be cut (Rmin, Rmax).

Read a model
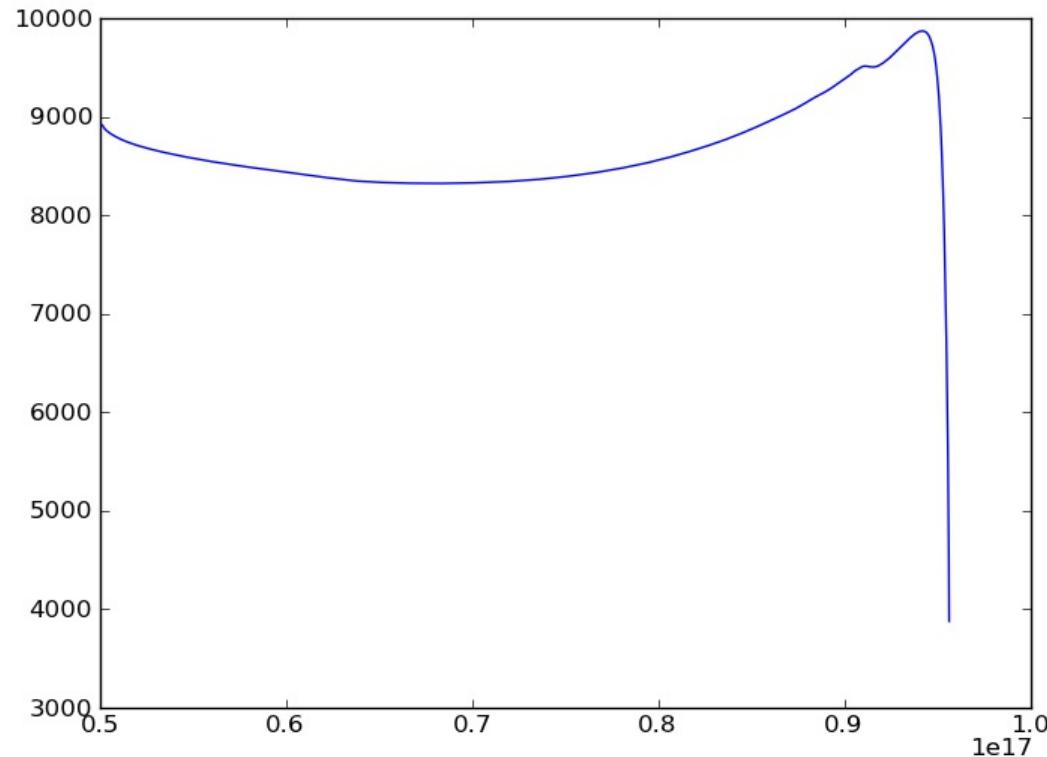

```
M = pc.CloudyModel('Models/model2')

plt.plot(M.radius, M.te)
```

Plot model results
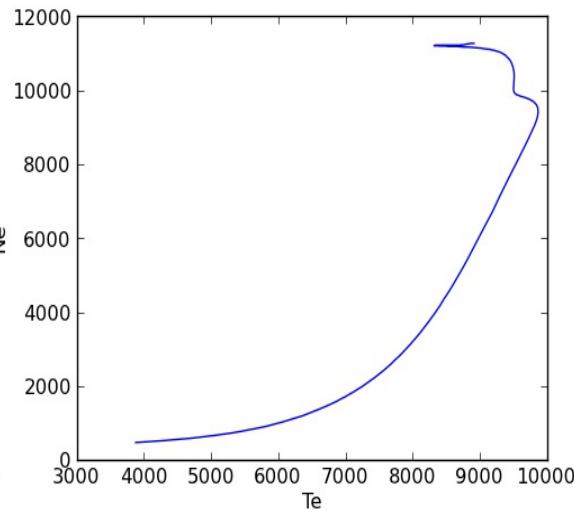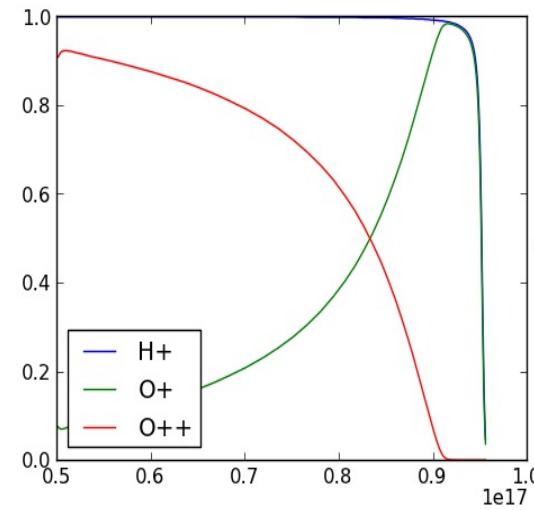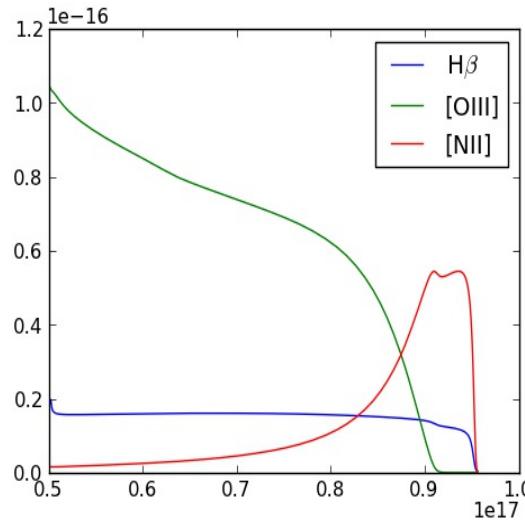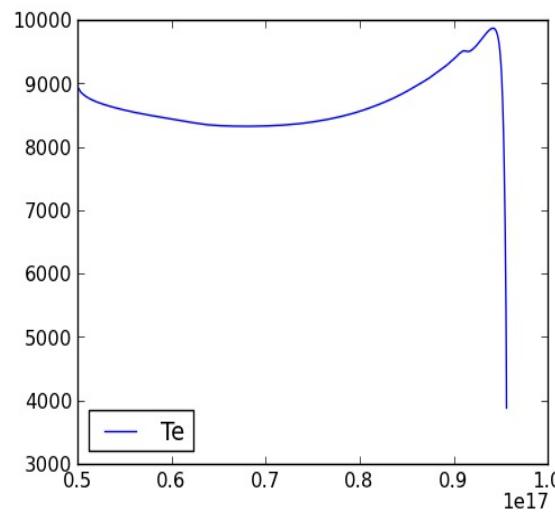

```
plt.subplot(2,2,1)

plt.plot(M.radius, M.te, label = 'Te')

plt.legend(loc=3)

plt.subplot(2,2,2)

plt.plot(M.radius, M.get_emis('H__1__4861A'), label = r'H$\beta$')

plt.plot(M.radius, M.get_emis('O__3__5007A'), label = '[OIII]')

plt.plot(M.radius, M.get_emis('N__2__6584A'), label = '[NII]')

plt.legend()

plt.subplot(2,2,3)

plt.plot(M.radius, M.get_ionic('H', 1), label = 'H+')

plt.plot(M.radius, M.get_ionic('O', 1), label = 'O+')

plt.plot(M.radius, M.get_ionic('O', 2), label = 'O++')

plt.legend(loc=3)

plt.subplot(2,2,4)

plt.plot(M.te, M.ne)

plt.xlabel('Te')

plt.ylabel('Ne')
```

Plot continuum
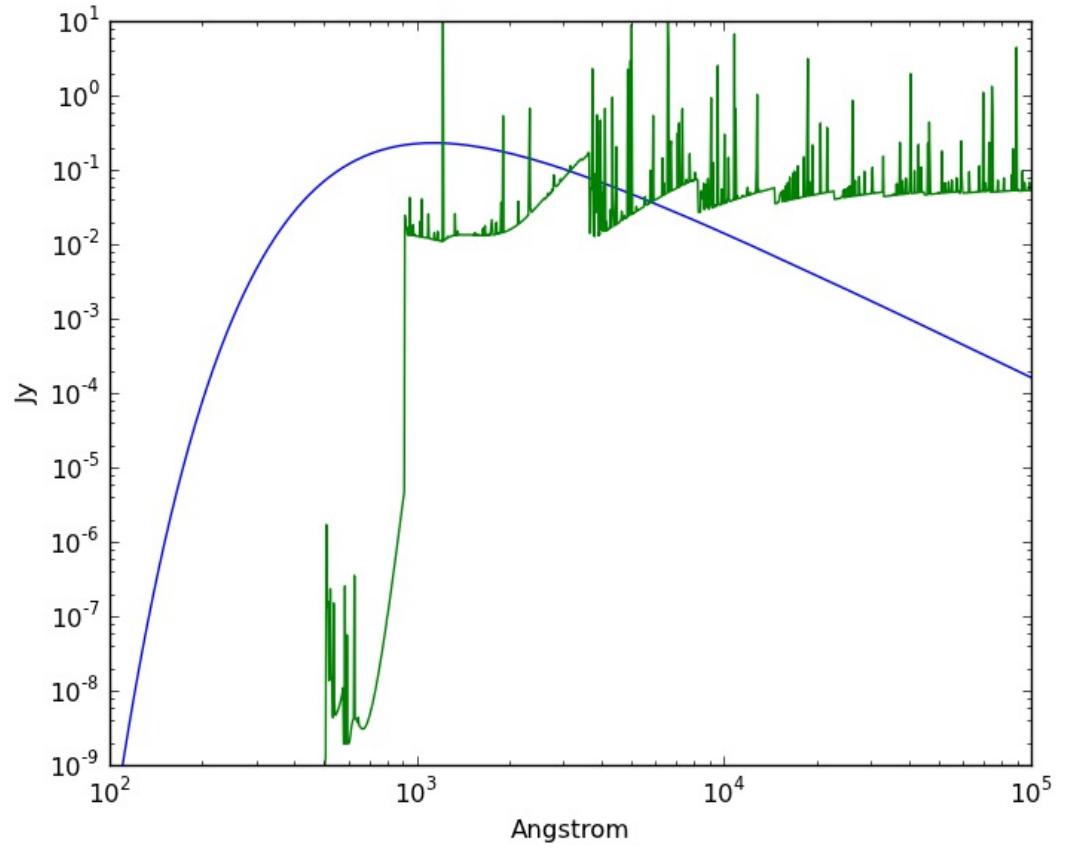

```
M.distance = 2.3 #kpc

plt.loglog(M.get_cont_x(unit='Ang'), M.get_cont_y(cont = 'incid', unit = 'Jy'), label = 'Incident')

plt.loglog(M.get_cont_x(unit='Ang'), M.get_cont_y(cont = 'diffout', unit = 'Jy'), label = 'Diff Out')

plt.xlim((100, 100000))

plt.ylim((1e-9, 1e1))

plt.xlabel('Angstrom')

plt.ylabel('Jy')
```

Print some stats

In [13]: M.print_stats() #This method can be changed by the user...

Name of the model: /Users/christophemorisset/Cloudy_test/Models//model2

R_in (cut) = 5.012e+16 (5.012e+16), R_out (cut) = 9.561e+16 (9.561e+16)

H+ mass = 2.56e-02, H mass = 2.62e-02

<H+/H> = 0.99, <He++/He> = 0.00, <He+/He> = 0.85

<O+++/O> = 0.00, <O++/O> = 0.54, <O+/O> = 0.44

T(O+++) = 8815, T(O++) = 8520, T(O+) = 9059

<log U> = -2.34

Print some stats (2)

In [16]: M.get_T0_emis('H__1__4861A')

Out[16]: 8743.8

In [17]: M.get_t2_emis('H__1__4861A')

Out[17]: 0.00264

In [18]: M.get_ab_ion_vol_ne('O', 2)

Out[18]: 0.54408

In [19]: M.get_T0_ion_vol_ne('O', 2)

Out[19]: 8519.5

In [20]: M.log_U_mean

Out[20]: -2.342

In [21]: M.log_U[0]

Out[21]: -1.976

In [22]: M.log_U[-1]

Out[22]: -2.537

Grid of models

- It's very easy to define and run a grid of models.

- As it is in a programmatic language, it's very easy to define and write a set of input files (grid).

- The « make » facility is used to run N input files at the same time. This can be called from python.

- It's also easy to read the results of the whole grid and to play with them.

Grid of models


```python
tab_dens = [3, 4, 5, 6]

tab_ab_O = [-3.1, -3.25, -3.4, -3.55, -3.7]

for dens in tab_dens:

    for ab_O in tab_ab_O:

        make_model(dir_, model_name, dens, ab_O)

c_input.run_cloudy(dir_ = dir_, n_proc = 3, model_name = model_name)

Ms = pc.load_models('{0}/{1}'.format(dir_, model_name), read_grains = False)

rO3 = [np.log10(M.get_emis_vol('O__3__5007A')/M.get_emis_vol('TOTL__4363A')) for M in Ms]

rO2 = [np.log10(M.get_emis_vol('O_II__3726A')/M.get_emis_vol('O_II__3729A')) for M in Ms]

col = [M.abund['O'] for M in Ms]

size = [np.log10(M.nH[0])*20 for M in Ms]

plt.scatter(rO2, rO3, c=col, s=size)

plt.xlabel('log [OII] 3726/3729')

plt.ylabel('log [OIII] 5007/4959')
```
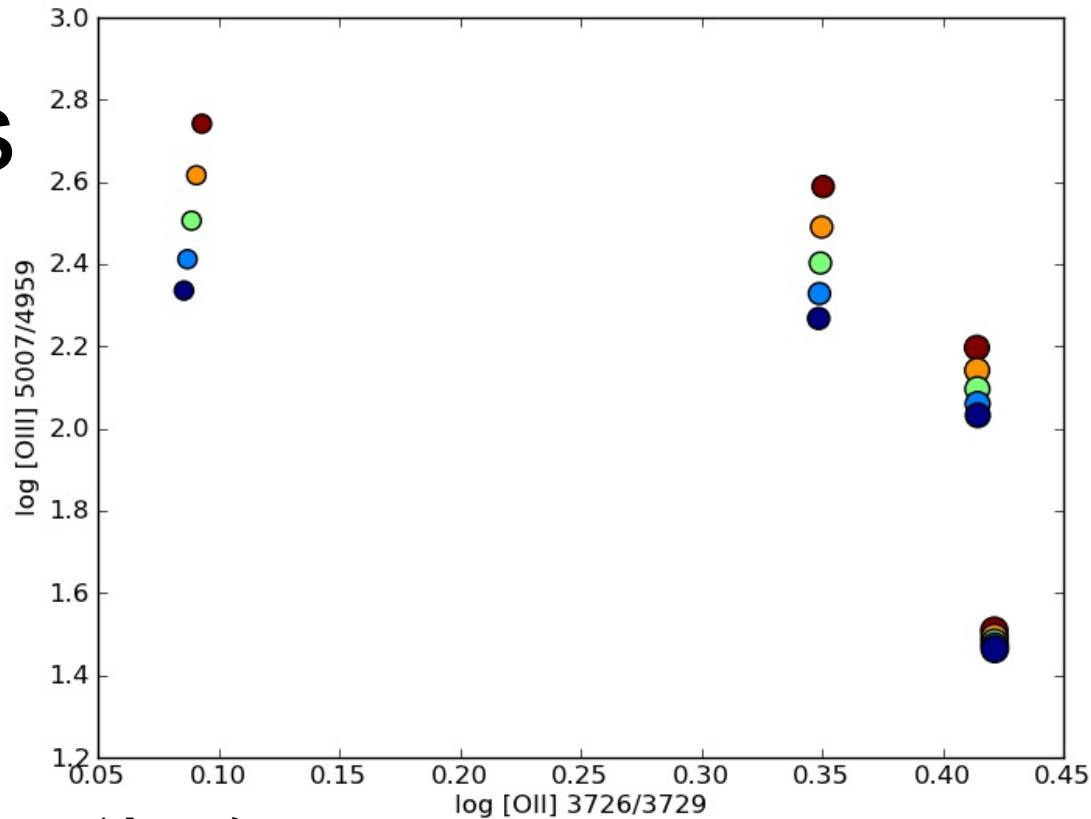
Grid of models
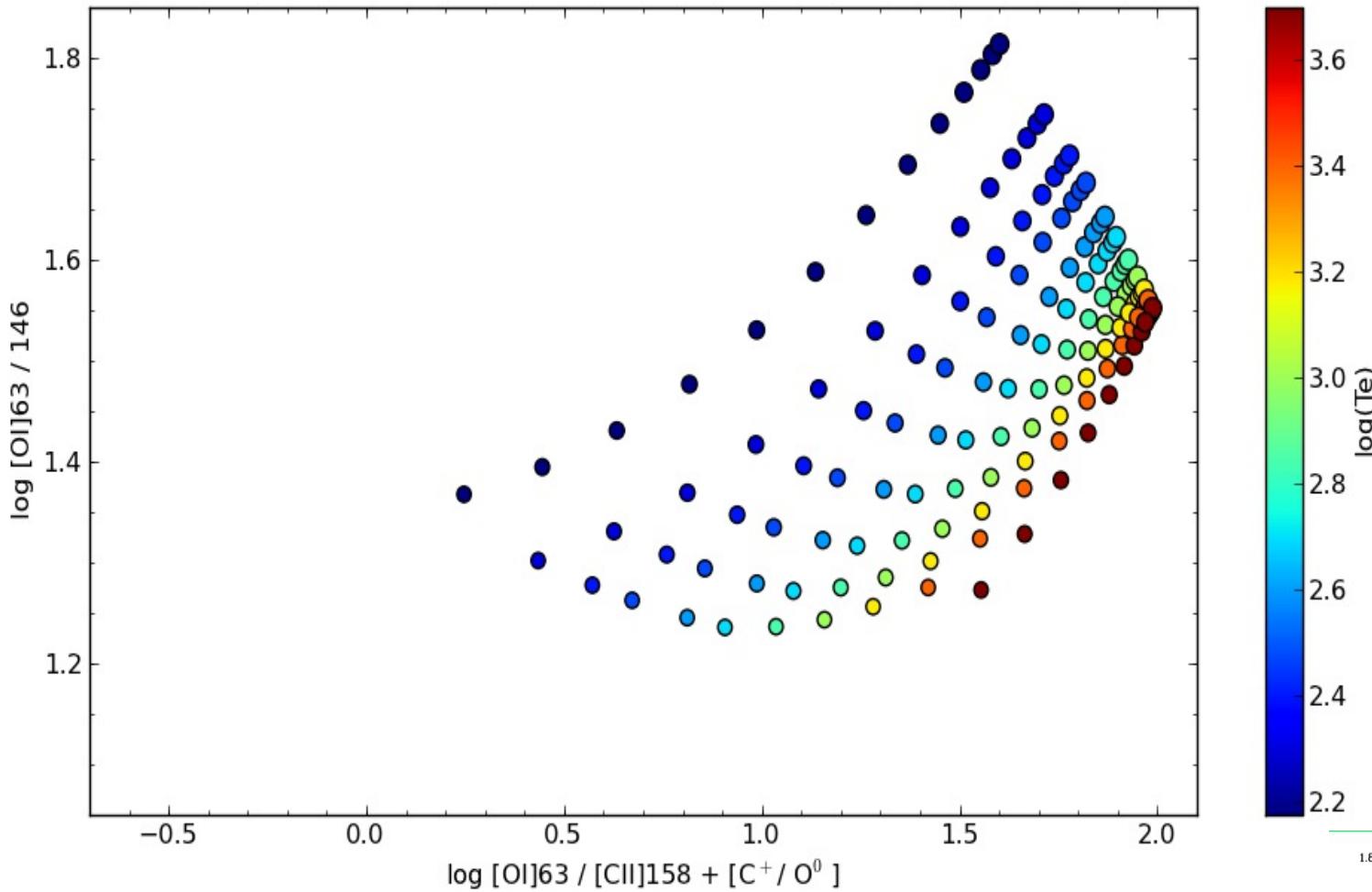
Grid of 1-zone models to explore diagnostic line ratios.

Compare to Liu et al.

ISO LWS observations of planetary nebula fine-structure lines

X.-W. Liu,[1★] M. J. Barlow,[1] M. Cohen,[2] I. J. Danziger,[3] S.-G. Luo,[4] J. P. Baluteau,[5] P. Cox,[6] R. J. Emery,[7] T. Lim[7] and D. Péquignot[8]
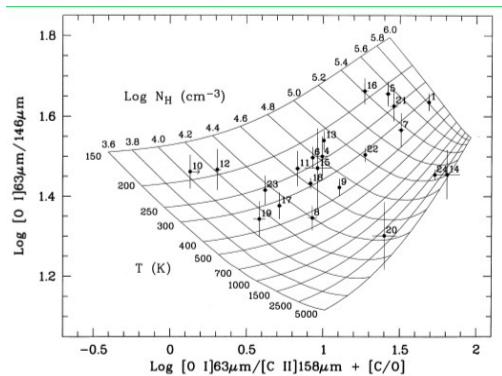
3D models

- A grid of 1D models with varying « theta » and « phi » parameters is run.

- A 3D cube of coordinates (R, theta, phi) is generated. Rotation may apply.

- For each spaxel of the coordinate cube, the values of Te, Xi/X, emissivities can be interpolated from the 2 or 3 closest 1D runs.

3D models

- The main « difficulty » is to define the morphology. What and how things change with theta (and phi, not always needed) ?

- Needs to have a clear 3D view of the object.

- Sometimes one can use a topological equivalent shape, no need for the detailed model (e.g. metal-rich clumps).

3D models
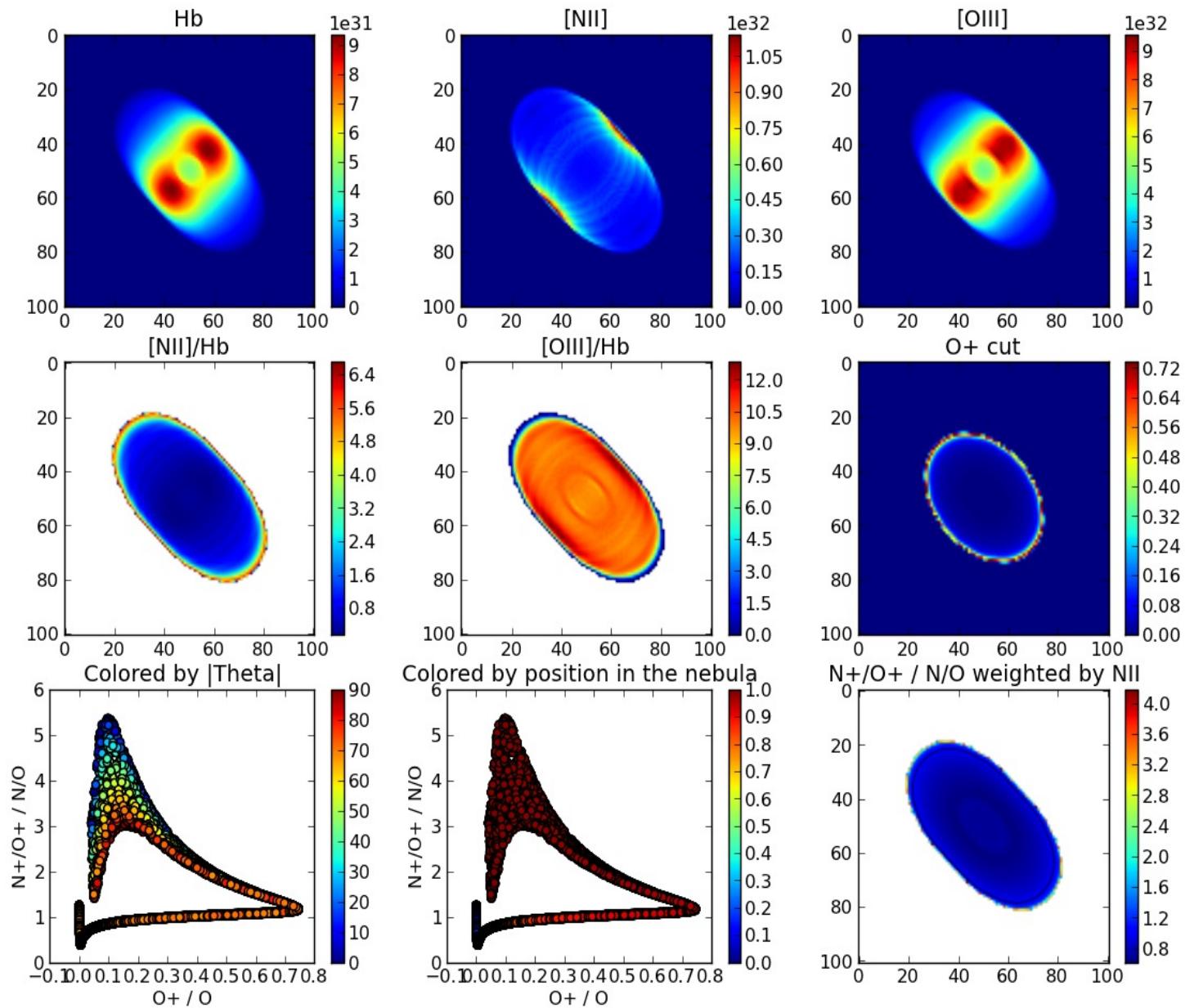
3D models
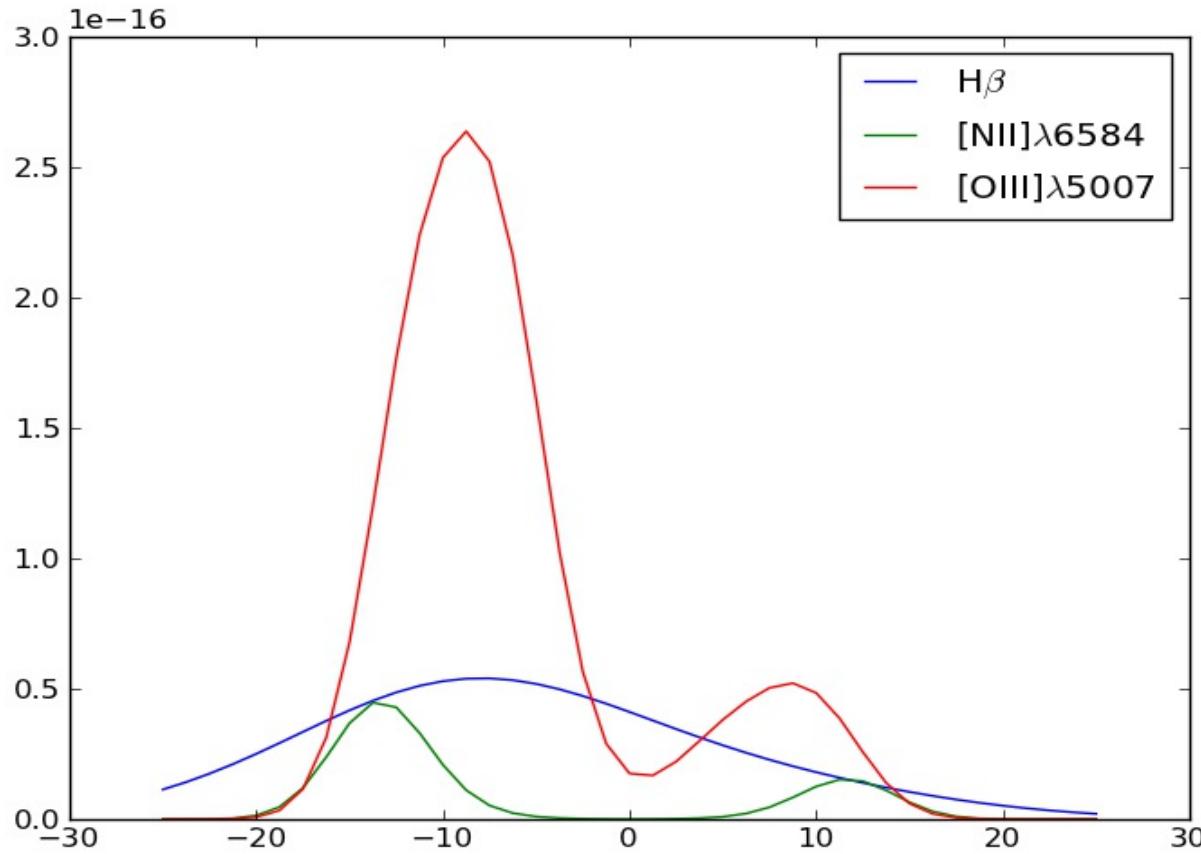
- A velocity field can be defined and line profiles are computed, taking into account thermal and turbulent broadening.
- The profiles can be « observed » using a mask (slit+seeing).