



EUROPEAN ARC

ALMA Regional Centre || Allegro

# Monte Carlo methods for molecular line observations at (sub)mm wavelengths

Michiel Hogerheijde  
Leiden Observatory  
Allegro ALMA ARC node

# Outline

- Lecture 1:
  - How does molecular line transfer and non-LTE excitation work?
- Lecture 2:
  - How can Monte Carlo methods help?
  - How does LIME work?
  - Practical work with LIME Q&A (tomorrow)



EUROPEAN ARC

ALMA Regional Centre || Allegro



# How can Monte Carlo methods help? How does LIME work?

Michiel Hogerheijde  
Leiden Observatory  
Allegro ALMA ARC node

# Outline

- How Monte Carlo methods can help
- The trouble with Monte Carlo methods
- By-passing some of these problems:  
accelerated methods
- Bernes, RATRAN, and LIME
- How to work with LIME

# Outline

- ➔ How Monte Carlo methods can help
- The trouble with Monte Carlo methods
  - By-passing some of these problems:  
accelerated methods
  - Bernes, RATRAN, and LIME
  - How to work with LIME

# Recap of RT

- Intensity along a ray is increased by emission and decreased by absorption

$$\frac{dI_\nu}{ds} = -\alpha_\nu I_\nu + j_\nu$$

- Rewrite as

$$\frac{dI_\nu}{d\tau_\nu} = -I_\nu + S_\nu$$

$$d\tau_\nu \equiv \alpha_\nu ds$$

$$S_\nu \equiv \frac{j_\nu}{\alpha_\nu}$$

- Emission and absorption takes place within a given frequency response ('line profile function'), including contributions from the intrinsic line width, thermal, and (micro) turbulent broadening

# Recap of RT (2)

- Emission and absorption coefficients are given by
  - the Einstein coefficient for spontaneous emission  $A_{ij}$
  - the derived coefficients for stimulated emission and absorption  $B_{ij}$  and  $B_{ji}$ .

$$g_1 B_{12} = g_2 B_{21}$$

$$A_{21} = \frac{2h\nu^3}{c^2} B_{21}.$$

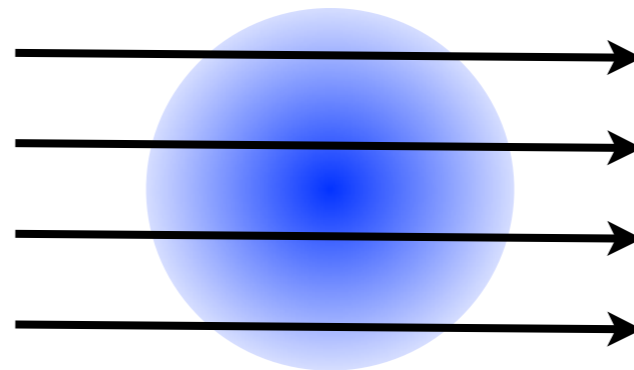
- and the populations of the involved energy levels  $n_{ij}$

$$j_\nu = \frac{h\nu_0}{4\pi} n_2 A_{21} \phi(\nu)$$

$$\alpha_\nu = \frac{h\nu_0}{4\pi} (n_1 B_{12} - n_2 B_{21}) \phi(\nu).$$

# A coupled problem

- The eqn of statistical equilibrium includes the average radiation field  $J$
- In the limit that the radiation field generated by the atoms / molecules is weak, the solution is straightforward
  - optically thin limit
  - only include the strength of the CMB and any dust continuum field
  - neither depends on the level populations: the problem is *local*
- $\Rightarrow$  calculate excitation based on collisions and  $J$
- $\Rightarrow$  ray-tracing gives  $I_\nu(\alpha, \delta)$





# A coupled problem (2)

- In the limit that the radiation field generated by the atoms / molecules is not negligible, the problem becomes coupled
  - optically thick limit
  - $J$  follows from solving the RT eqn including line photons

$$n_l \left[ \sum_{k < l} A_{lk} + \sum_{k \neq l} (B_{lk} J_\nu + C_{lk}) \right] = \sum_{k > l} n_k A_{kl} + \sum_{k \neq l} n_k (B_{kl} J_\nu + C_{kl}).$$

&

iterate

$$\frac{dI_\nu}{ds} = -\alpha_\nu I_\nu + j_\nu$$

$$j_\nu = \frac{h\nu_0}{4\pi} n_2 A_{21} \phi(\nu)$$

$$\alpha_\nu = \frac{h\nu_0}{4\pi} (n_1 B_{12} - n_2 B_{21}) \phi(\nu).$$

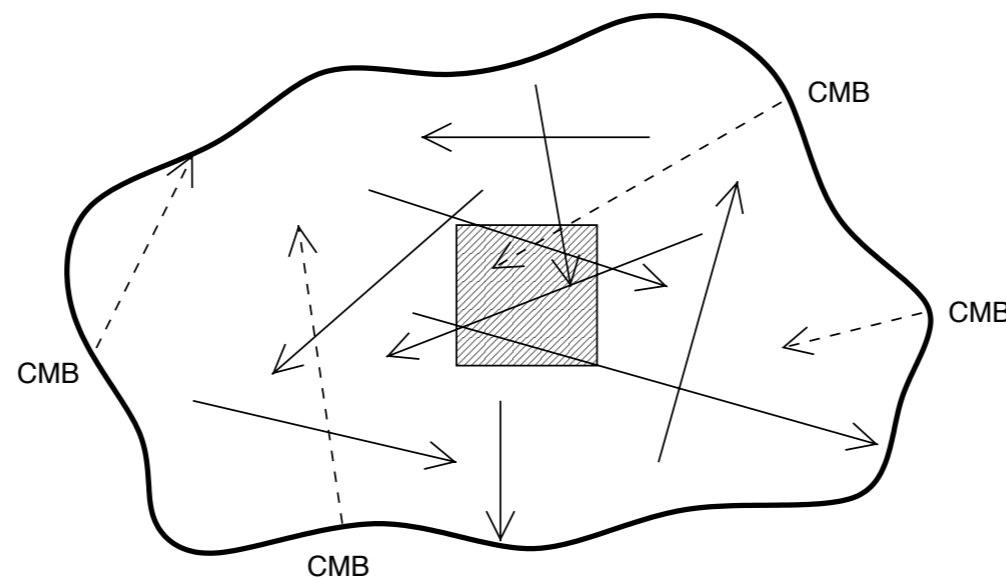
$$J_\nu \equiv \frac{1}{4\pi} \int I_\nu d\Omega$$

# Solution methods

- In principle, one can do a *full* calculation for the excitation at *all* locations in your object, solving RT along *all* lines of sight, and iteratively obtaining a solution
- In practice, this is impractical
- So either
  - perform this iterative calculation for a *limited set of locations* in the object and a *limited set of rays*
    - e.g., Monte Carlo methods, Accelerated Lambda Iteration
  - or use an *approximation to the long-range RT*, and only self-consistently calculate the *local* radiative coupling
    - e.g., Large Velocity Gradient / Sobolev, Escape Probability
- Many hybrid methods exist, that work well for particular situations

# Example: Monte Carlo methods

- If you suspect that a local approximation like an escape probability is not good enough for your problem, you need to use an iterative method including ‘full’ sampling of the RT, or at least a sufficiently good approximation of that
- Methods: Monte Carlo methods (RATRAN, LIME, ...); Accelerated Lambda Iteration, ...
- Penalty: (much!) longer calculation times
- Bonus: more accurate line *profiles*

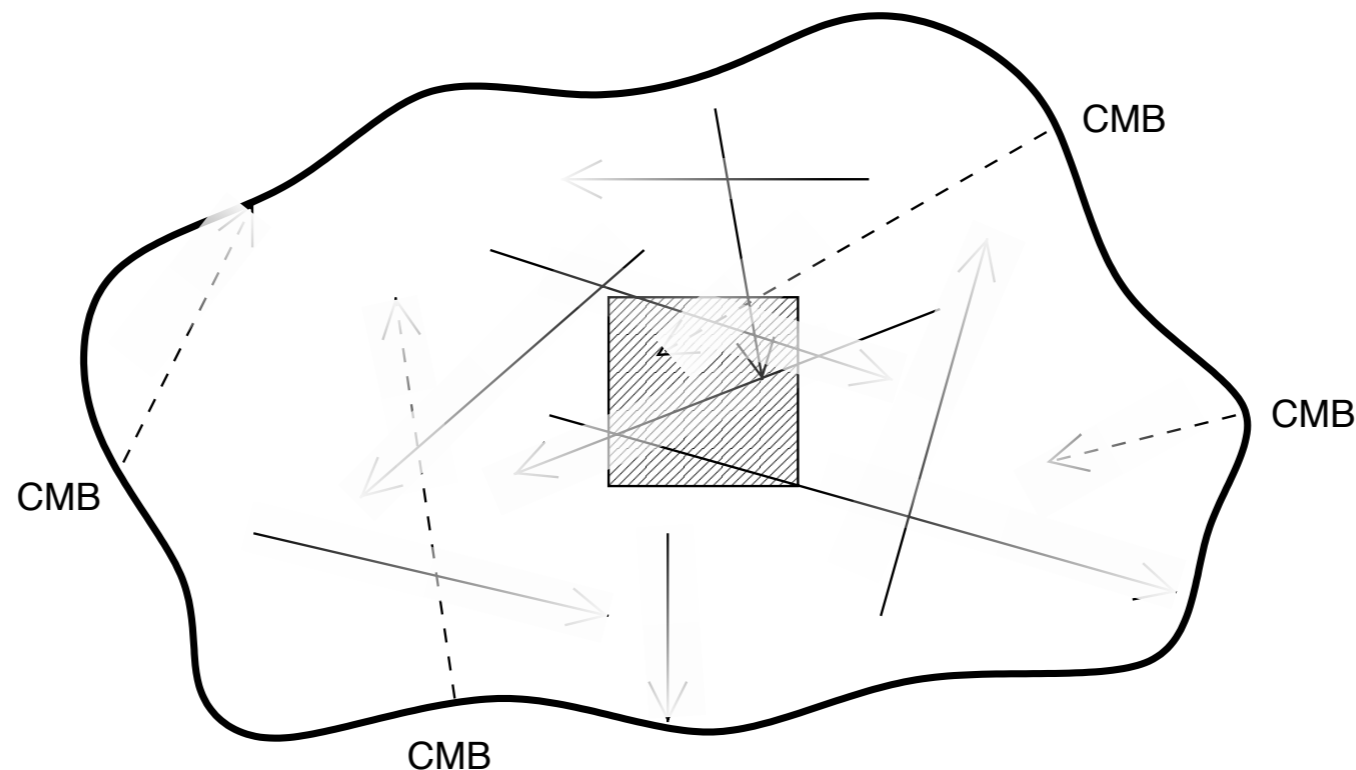


# Outline

- How Monte Carlo methods can help
- ➔ **The trouble with Monte Carlo methods**
- By-passing some of these problems:  
accelerated methods
- Bernes, RATRAN, and LIME
- How to work with LIME

# The trouble with MC

- If (line) opacities become large, information is rapidly destroyed along a photon propagation
- This means that high excitation only travels slowly
- Slow change in excitation is misinterpreted as convergence!
- This is a common problem, not limited to MC methods



# A math solution (1)

- In ‘Lambda Iteration’ we write the coupled problem of radiative transfer and molecular excitation in the following form

$$J_\nu = \Lambda [S_{ul}(J_\nu)]$$

- In other words: the average radiation field is found by having an operator  $\Lambda$  work on the source function  $S$ , which in it itself is a function of the radiation field  $J$ .
- $\Lambda$  describes the RT ‘connection’ between all grid points.
- In principle, one can find the solution of this equation by inverting  $\Lambda$ . However, inverting a matrix is an expensive operation. So it is easier to find a solution iteratively

$$J_\nu = \Lambda[S_{ul}^\dagger(J_\nu)]$$

- Here  $S^\dagger$  is the previous excitation solution.

# A math solution (2)

- This iterative solution suffers from the same problem as MC: if opacities are large, the terms in  $\Lambda$  allow very little change in the solution from iteration to iteration, which can be misunderstood as convergence.
- Rybicki & Hummer (1991) suggested a method called ‘accelerated Lambda iteration’ (ALI), by splitting  $\Lambda$  in a diagonal ( $\Lambda^*$ ) and a non-diagonal ( $\Lambda - \Lambda^*$ ) part

$$J_\nu = (\Lambda - \Lambda^*) [S_{ul}^\dagger(J_\nu)] + \Lambda^* [S_{ul}(J_\nu)]$$

- Because a diagonal matrix can be easily inverted, the next iteration is found by the combination of the non-diagonal part working on the previous solution, and the *true* solution of the diagonal part working on  $S$ .
- This converges much faster.

# A math solution (3)

- ALI converges much faster than regular LI in the presence of large opacity.
- This is a math trick, but it must have a root in physics.
  
- What is the diagonal part of the operator?
  - It is the interaction of the grid point with itself
  - which, in the presence of large opacity, is largely (but not entirely) responsible for the local radiation field.
  - So finding the full solution to this ‘self-interaction’ fast, speeds up convergence!

▶ **Aha, we can use that for MC as well!**



# Outline

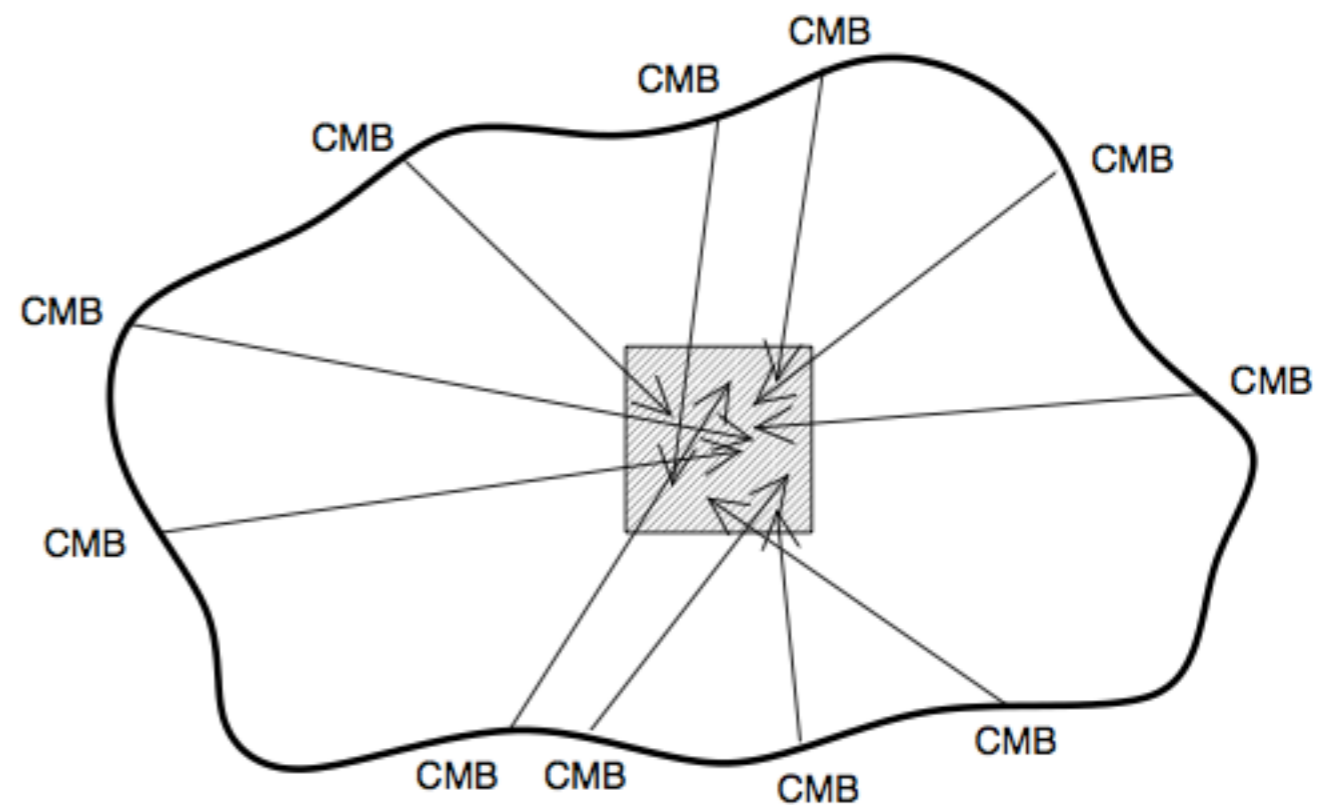
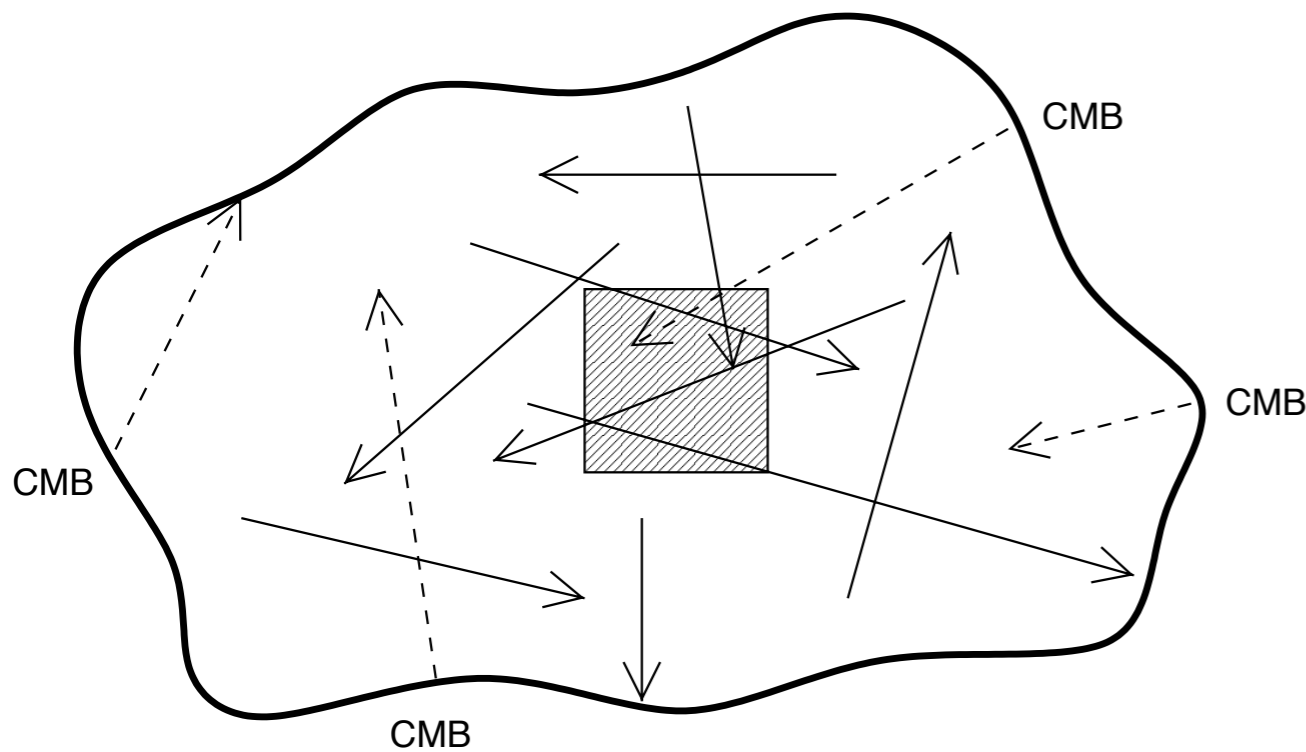
- How Monte Carlo methods can help
- The trouble with Monte Carlo methods
- ➔ **By-passing some of these problems:  
accelerated methods**
- Bernes, RATRAN, and LIME
- How to work with LIME

# Example: Monte Carlo methods

- Random photon packages make it easy to separate the locally produced radiation field from the global field.
- But you can think of MC as a method to solve the *integral*

$$J_\nu \equiv \frac{1}{4\pi} \int I_\nu d\Omega$$

- Now it is much easier to separate the two radiation fields!



# Outline

- How Monte Carlo methods can help
  - The trouble with Monte Carlo methods
  - By-passing some of these problems:  
accelerated methods
- ➔ **Bernes, RATRAN, and LIME**
- How to work with LIME

# Bernes 1979

- First MC code for molecular line RT
- FOTRAN77, with numerical routines from IBM

Astron. Astrophys. 73, 67–73 (1979)

ASTRONOMY  
AND  
ASTROPHYSICS

## A Monte Carlo Approach to Non-LTE Radiative Transfer Problems

C. Bernes

Stockholms Observatorium, S-13300 Saltsjöbaden, Sweden

Received May 25, revised August 10, 1978

**Summary.** A procedure for the solution of non-LTE, multi-level, radiative transfer problems using the Monte Carlo method is developed and shown to be useful. This procedure allows the inclusion of velocity fields, inhomogeneities and complex geometries. It is applied to a study of the transfer of CO line radiation in a model of a spherical, collapsing dark cloud. A technique for reducing the random fluctuations inherent in the Monte Carlo method is introduced. This technique typically cuts down computing times by an order of magnitude in the investigated case. Line profiles accurate to 1% or better were obtained with a very reasonable amount of computing time.

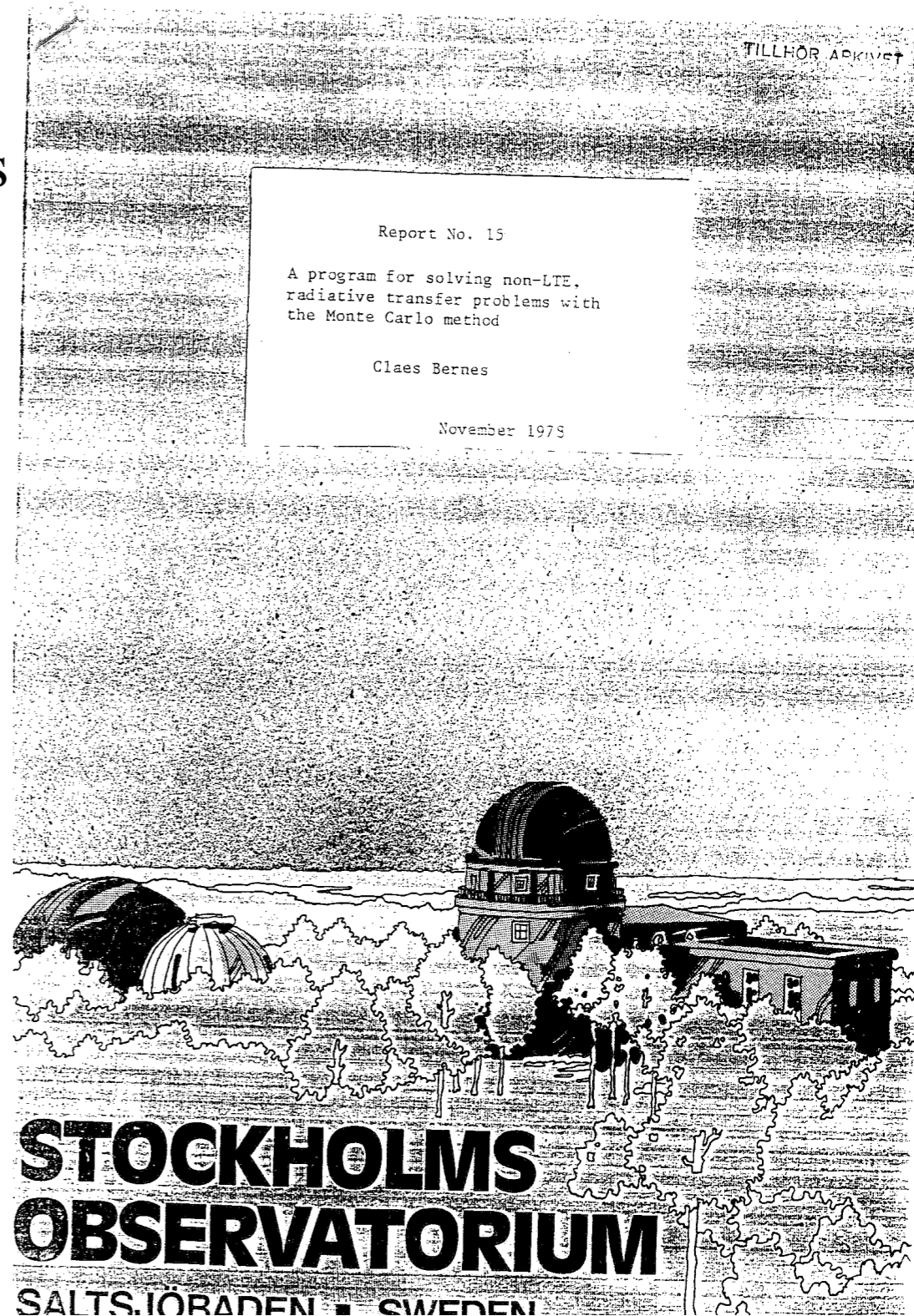
**Key words:** radiative transfer – Monte Carlo methods – dark clouds – interstellar molecules – carbon monoxide

### 1. Introduction

Conventional numerical methods for solution of multi-level, non-LTE, radiative transfer problems have been developed to a high

lines (cf. Auer, 1968; Avery and House, 1968; Avery et al., 1969; Caroff et al., 1972; Magnan, 1970 and 1972; Modali et al., 1972 and 1975), but none of these works contains a self-consistent non-LTE analysis or a study of a multi-level system. Fleck (1963) has described a general Monte Carlo procedure for the calculation of non-linear radiative transfer that is well adapted to studies of time-dependent systems. It is similar in some basic respects to the method developed here, but requires more memory space. Fleck did not apply the procedure specifically to line radiation, but noted that this could be done. Price (1969) made an elaborate non-LTE analysis of a pure hydrogen stellar atmosphere, and included continuous opacity sources, but treated the radiation in the lines in a crude way.

The aims of this paper are to demonstrate the simplicity of the Monte Carlo approach to non-LTE, multi-level, radiative transfer problems, and to show that the method is worthy of consideration in studies of line radiation in such regions as extended stellar envelopes and interstellar clouds. The procedure is described in detail in Sect. 2. A technique for variance-reduction



10.

Program listing

```

C THIS PROGRAM SOLVES NON-LTE, MULTI-LEVEL, RADIATIVE TRANSFER
C PROBLEMS IN SPHERICAL GEOMETRY WITH THE MONTE CARLO METHOD.
C
C THE RADIATION IN ALL RADIATIVE TRANSITIONS IS SIMULTANEOUSLY RE-
C PRESENTED BY A NUMBER OF MODEL PHOTONS.
C THE SPHERICAL REGION IS DIVIDED INTO SPHERICAL SHELLS, NUMBERED
C OUTWARDS FROM 1 TO NSHELL. TO EACH SHELL IS ASSIGNED A KINETIC
C TEMPERATURE (TK), A COLLISION AGENT NUMBER DENSITY (DENS) AND A
C MICROTURBULENT VELOCITY (VTURB).
C A RADIAL VELOCITY FIELD MAY BE SPECIFIED BY THE FUNCTION V(R).
C THE ENERGY LEVELS UNDER CONSIDERATION ARE NUMBERED FROM 1 (THE
C LOWEST) TO NLEVEL (THE HIGHEST). TRANSITIONS BETWEEN ANY TWO
C DIFFERENT ENERGY LEVELS MAY BE TREATED AS RADIATIVE BY THE PRO-
C GRAM. ARRAY LTRANS SPECIFIES WHICH LEVELS ARE CONNECTED IN THIS
C WAY. A CONTINUOUS EXTERIOR RADIATION FIELD, SPECIFIED BY RADIA-
C TION TEMPERATURE TBG, AND CONTINUOUS RADIATION FROM A CENTRAL
C REGION, SPECIFIED BY RADIATION TEMPERATURE TCEN, CAN ALSO BE
C INCLUDED. IT WILL IN MANY CASES BE ADVANTAGEOUS TO APPLY A REFER-
C ENCE FIELD WITH RADIATION TEMPERATURE TREF AS AN APPROXIMATION
C OF THE TRUE FIELD IN THE REGION.
C THE SIMULATION OF THE RADIATION IN THE REGION IS CARRIED OUT
C THROUGH A NUMBER (NITER) OF ITERATIONS, EACH CONSISTING OF NPHOT
C MODEL PHOTON EMISSIONS. THE NUMBER OF INDUCED RADIATIVE TRANSI-
C TIONS IS MEASURED BY THE SSTIM COUNTERS DURING THESE ITERATIONS,
C AND USED TO DERIVE CONSISTENT LEVEL POPULATIONS. THE COUNTERS
C MAY BE EMPTIED AFTER ANY ITERATION(S) - CF. ARRAY NITEMP. WHEN
C ALL ITERATIONS ARE COMPLETED, SUBROUTINE PROFIL CALCULATES THE
C EMITTED SPECTRUM.
C CGS UNITS ARE USED ALMOST EXCLUSIVELY; THE ONLY EXCEPTIONS ARE
C THE PASSBAND PASSB, THE MICROTURBULENT VELOCITIES VTURB AND THE
C VELOCITY FIELD V(R) (AND A FEW CONSTANTS RELATED TO THESE
C VALUES), WHICH ARE GIVEN IN KM/S, AND THE ATOMIC WEIGHT AMASS,
C WHICH IS GIVEN IN ATOMIC UNITS.
C
C THE MAIN PROGRAM CONTROLS THE FLOW OF THE EXECUTION.
C
C SUBROUTINE INITIAL READS INPUT VALUES, DEFINES CONSTANTS AND PRE-
C PARES ARRAYS FOR FUTURE USE.
C
C SUBROUTINE RUN PERFORMS THE MONTE CARLO SIMULATION OF THE RADI-
C ATIVE TRANSFER. IT CALCULATES THE WEIGHTS OF THE MODEL PHOTONS
C TO BE EMITTED IN THE VARIOUS SHELLS AND INJECTED FROM THE CENTRAL
C REGION AND FROM THE OUTSIDE, AND MEASURES THE NUMBER OF RADI-
C ATION-INDUCED TRANSITIONS.
C
C SUBROUTINE STEP COMPUTES FOR EACH STEP OF THE MODEL PHOTONS THE
C DISTANCE FROM THE CENTRE (R), THE COSINE OF THE ANGLE BETWEEN
C THE PHOTON DIRECTION AND THE RADIAL DIRECTION (U) AND THE STEP
C LENGTH (DS).
C
C SUBROUTINE ADJUST PREPARES FOR THE SOLUTION OF THE STATISTICAL
C EQUILIBRIUM EQUATIONS. THE ELEMENTS OF THE MATRICES (A) THAT
C ARE TO BE INVERTED ARE CALCULATED. THE SOLUTIONS ARE PRINTED.
C
C SUBROUTINE INVERT INVERTS THE A MATRICES.

```

```

SUBROUTINE INVERT(M,IER)
C
C WHEN SUBROUTINE INVERT IS CALLED, THE PN(M,..) ROW CONTAINS THE
C RIGHT-HAND SIDES OF THE STATISTICAL EQUILIBRIUM EQUATIONS. ON RE-
C TURN IT CONTAINS THE SOLUTIONS OF THE EQUATIONS. THE ERROR PARA-
C METER IER IS CODED AS FOLLOWS: IER=0 - NO ERROR. IER=-1 -
C NO RESULT BECAUSE OF NLEVEL LESS THAN 1 OR PIVOT ELEMENT AT ANY
C ELIMINATION STEP EQUAL TO 0. IER=K - WARNING DUE TO POSSIBLE
C LOSS OF SIGNIFICANCE INDICATED AT ELIMINATION STEP K+1.
C
C THIS SUBROUTINE IS ADAPTED FROM SUBROUTINE GELG IN THE IBM SSP-
C LIBRARY, WHICH MAY BE CONSULTED FOR FURTHER DOCUMENTATION.
C
COMMON /ALL/ FREQ(5),GG(5),LTRANS(5,2),PN(15,6),HK,NTRANS
COMMON /RUNADJ/ A(6,6),C(15,6,6),DENS(15),F(5),NITEMP(9),
- SSTIM(15,5),VOL(15),EPS,FRAC,NEMPTY,NLEVEL,NLEV1,NPHOT,
- NSHELL,SEED,TPI,TRE
DOUBLE PRECISION SEED
IF (NLEVEL) 23,23,1
1 IER=0
PIV=0.
MM=NLEVEL*NLEVEL
DO 3 L=1,MM
TB=ABS(A(L))
IF (TB-PIV) 3,3,2
2 PIV=TB
I=L
3 CONTINUE
TOL=EPS*PIV
LST=1
DO 17 K=1,NLEVEL
IF (PIV) 23,23,4
4 IF (IER) 7,5,7
5 IF (PIV-TOL) 6,6,7
6 IER=K-1
7 PIVI=1./A(I)
J=(I-1)/NLEVEL
I=I-J*NLEVEL-K
J=J+1-K
LL=K+I
TB=PIVI*PN(M,LL)
PN(M,LL)=PN(M,K)
PN(M,K)=TB
IF (K-NLEVEL) 9,18,18
9 LEND=LST+NLEVEL-K
IF (J) 12,12,10
10 II=J*NLEVEL
DO 11 L=LST,LEND
TB=A(L)
LL=L+II
A(L)=A(LL)
11 A(LL)=TB
12 DO 13 L=LST,MM,NLEVEL
LL=L+I
TB=PIVI*A(LL)
A(LL)=A(L)

```

Bernes' code served as the basis of many MC codes.

# RATRAN

- Hogerheijde & van der Tak (2000)
- Splitting the local and global radiation fields
- Sub-iterations on the local field & excitation speed up the code, because the correct solution to locally optically thick conditions is quickly found

Astron. Astrophys. 362, 697–710 (2000)

ASTRONOMY  
AND  
ASTROPHYSICS

## An accelerated Monte Carlo method to solve two-dimensional radiative transfer and molecular excitation

With applications to axisymmetric models of star formation

M.R. Hogerheijde<sup>1</sup> and F.F.S. van der Tak<sup>2</sup>

<sup>1</sup> Radio Astronomy Laboratory, University of California at Berkeley, Astronomy Department, 601 Campbell Hall # 3411, Berkeley, CA 94720-3411, USA (michi@astro.berkeley.edu)

<sup>2</sup> Sterrewacht Leiden, Postbus 9513, 2300 RA Leiden, The Netherlands (vdtak@strw.leidenuniv.nl)

Received 13 June 2000 / Accepted 21 July 2000

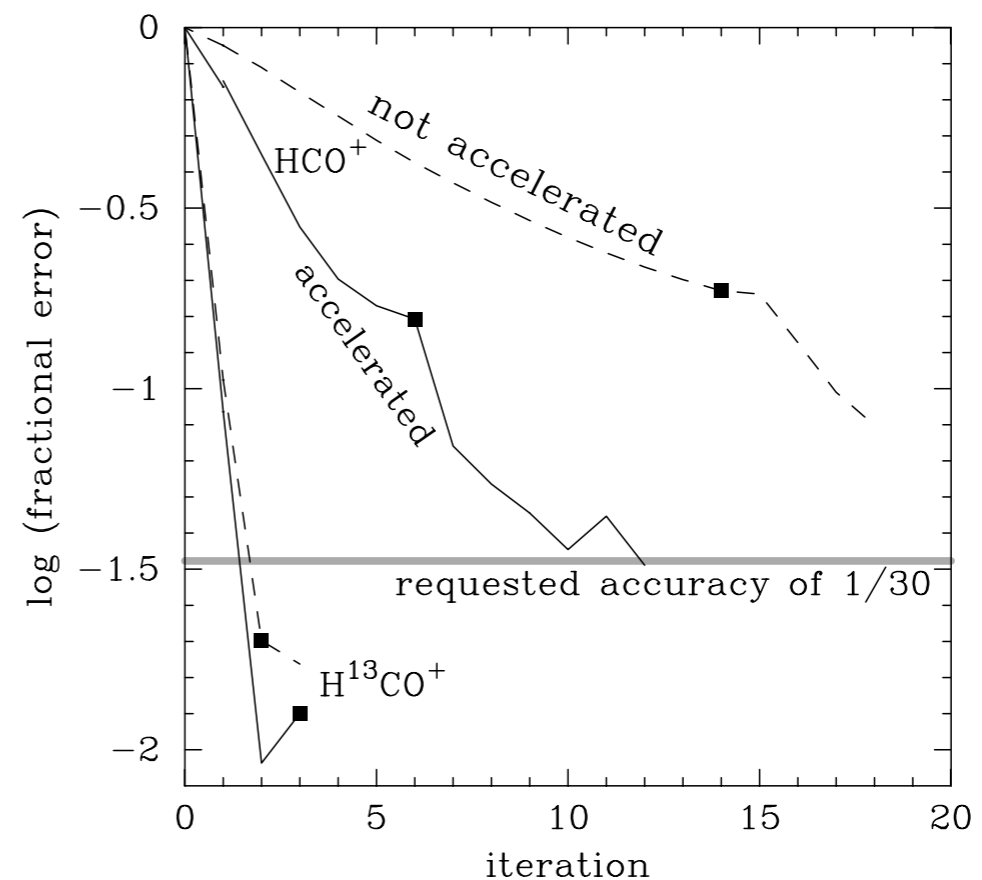
**Abstract.** We present a numerical method and computer code to calculate the radiative transfer and excitation of molecular lines. Formulating the Monte Carlo method from the viewpoint of cells rather than photons allows us to separate local and external contributions to the radiation field. This separation is critical to accurate and fast performance at high optical depths ( $\tau \gtrsim 100$ ). The random nature of the Monte Carlo method serves to verify the independence of the solution to the angular, spatial, and frequency sampling of the radiation field. These features allow use of our method in a wide variety of astrophysical problems without specific adaptations: in any axially symmetric source model and for all atoms or molecules for which collisional rate coefficients are available. Continuum emission and absorption by dust is explicitly taken into account but scattering is neglected. We illustrate these features in calculations of (i) the  $\text{HCO}^+$   $J=1-0$  and  $3-2$  emission from a flattened protostellar envelope with infall and rotation, (ii) the  $\text{CO}$ ,  $\text{HCO}^+$ ,  $\text{CN}$  and  $\text{HCN}$  emission from a protoplanetary disk and (iii)  $\text{HCN}$  emission from a high-mass young stellar object, where infrared pumping is important. The program can be used for optical depths up to  $10^3-10^4$ , depending on source model. We expect this program to be an important tool in analysing data from present and future infrared and (sub) millimetre telescopes.

**Key words:** line: formation – radiative transfer – methods: numerical – stars: formation – ISM: molecules

astrophysical problems. This is illustrated by the large number of infrared and submillimetre observatories planned for the near future, such as the Smithsonian Millimeter Array (SMA), the Atacama Large Millimeter Array (ALMA), the Far-Infrared and Submillimetre Space Telescope (FIRST) and the Stratospheric Observatory for Infrared Astronomy (SOFIA).

An essential step in the interpretation of the data from these instruments is the comparison with predicted emission from models. This paper presents a numerical method to solve the radiative transfer and molecular excitation in spherically symmetric and cylindrically symmetric source models. At the comparatively low densities of interstellar gas, the excitation of many molecules is out of local thermodynamic equilibrium (LTE), and the transfer of line (and continuum) radiation plays a significant role in determining the molecular excitation (Leung & Liszt, 1976; Black, 2000). Geometry thus becomes an important element, and the high spatial resolution of current and future instruments often demands that at least two-dimensional (axisymmetric) source structures are considered. In the implementation of our method discussed in this paper, we have limited the source structure to spherical and cylindrical symmetries. The large and often multidimensional parameter space further requires a fast and reliable method, which needs to be easily applicable to many different astrophysical problems.

This need for reliable and flexible tools calls for the use of Monte Carlo techniques, where the directions of integration are chosen randomly. This approach was first explored by Bernes (1979) for non-LTE molecular excitation; later, Choi et



# RATRAN

- Code consists of two steps
  - AMC: calculates excitation
    - output: level populations at all grid points
  - SKY: calculates image on the sky
    - for a source at given distance and orientation
    - output: FITS or MIRIAD image cube(s)

```

PROGRAM amc

c (c) Michiel Hogerheijde / Floris van der Tak 2000
c michiel@as.arizona.edu, vdtak@mpifr-bonn.mpg.de
c http://talisker.as.arizona.edu/~michiel
c http://www.mpifr-bonn.mpg.de/staff/fvandertak
c
c This file is part of the 'ratran' molecular excitation and
c radiative transfer code. The one-dimensional version of this code
c is publicly available; the two-dimensional version is available on
c collaborative basis. Although the code has been thoroughly tested,
c the authors do not claim it is free of errors or that it gives
c correct results in all situations. Any publication making use of
c this code should include a reference to Hogerheijde & van der Tak,
c 2000, A&A, 362, 697.
c
c mrh 01dec99 Fully revised version.
c fvdt 03sep03 New molecular data format
c
c Main program of amc.

IMPLICIT NONE
INCLUDE 'amccommon.inc'
INTEGER id,ilev,iter,niter,dummyi,newphot,conv,exceed,totphot,
$ totphot2,stage,fixseed
DOUBLE PRECISION var,goal,snr,minsnr,dummy,
$ opops(maxlev,maxcell),oopops(maxlev,maxcell),percent,avepops
$ ,ran1
CHARACTER*80 molfile,modelfile,outfile,collfile
EXTERNAL ran1

cc\
LOGICAL template
cc/

```

```

PROGRAM sky

c (c) Michiel Hogerheijde / Floris van der Tak 2000
c michiel@as.arizona.edu, vdtak@mpifr-bonn.mpg.de
c http://talisker.as.arizona.edu/~michiel
c http://www.mpifr-bonn.mpg.de/staff/fvandertak
c
c This file is part of the 'ratran' molecular excitation and
c radiative transfer code. The one-dimensional version of this code
c is publicly available; the two-dimensional version is available on
c collaborative basis. Although the code has been thoroughly tested,
c the authors do not claim it is free of errors or that it gives
c correct results in all situations. Any publication making use of
c this code should include a reference to Hogerheijde & van der Tak,
c 2000, A&A 362, 697.
c
c SKY generates MIRIAD images/cubes of the sky brightness, based
c on a model with levels populations and/or dust emissivity values.
c
c mrh 13jan00 Fully revised version
c
c Main program of sky.

IMPLICIT NONE
INCLUDE 'skycommon.inc'
CHARACTER filein*60,fileout*60,outname*60
INTEGER ix,iy,ixprime,ichan,iline,lout(maxrx)
INTEGER length
EXTERNAL length
REAL row(maxsky) ! REAL!!

```

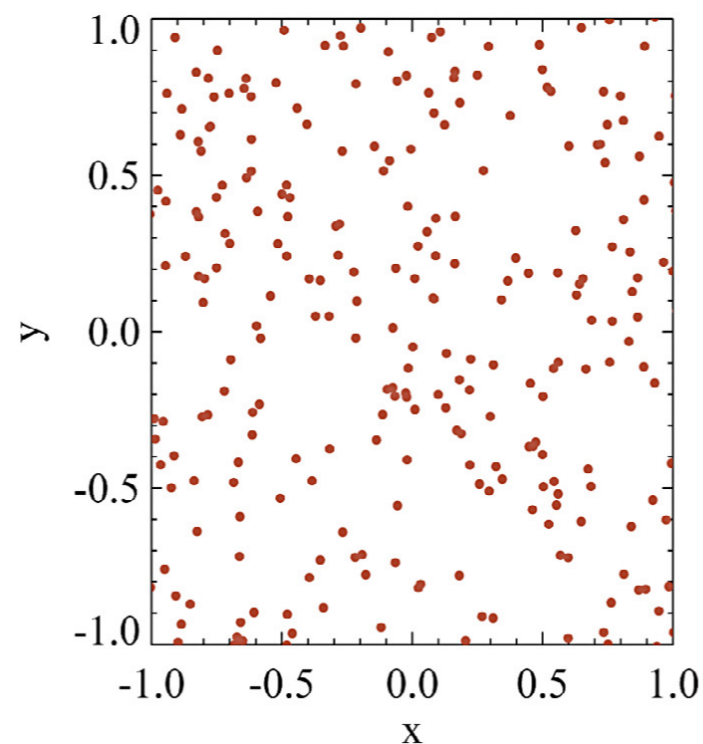
# RATRAN

- RATRAN requires a user-defined grid
  - in 1D: concentric shells
  - in 2D: nested cylinders
  - needs to fill entire space between  $r_{min}$  &  $r_{max}$  (and  $z_{min}$  &  $z_{max}$ )
  - needs to follow excitation gradient well
  - difficult to ‘see’ very small cells from distant cells: limits on dynamic range of spatial scales



# LIME

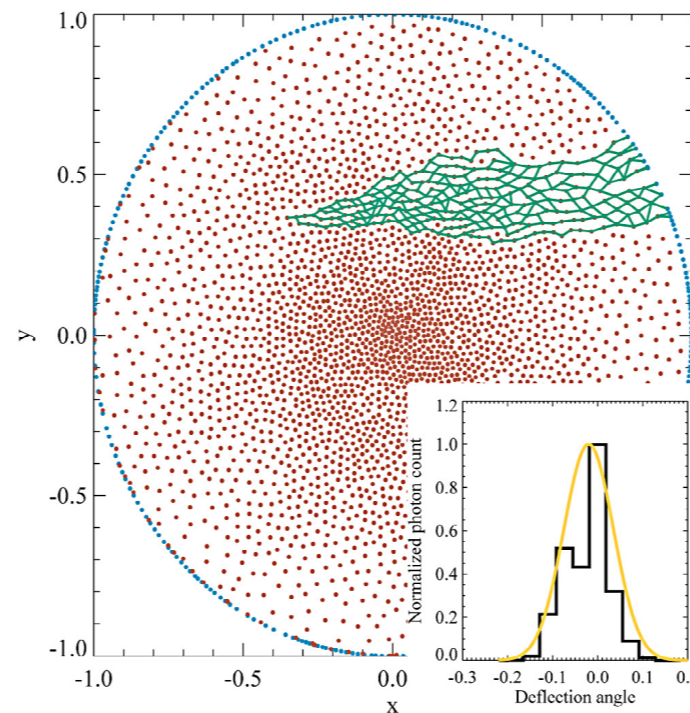
- LIME (Brinch & Hogerheijde 2010) was designed to overcome this problem
- fully 3D, using grid *points* instead of grid *cells*
- Place cells randomly with probability proportional to density<sup>0.2</sup>
  - mean separation proportional to opacity
  - other weighting schemes can be used



example for uniform density

# LIME

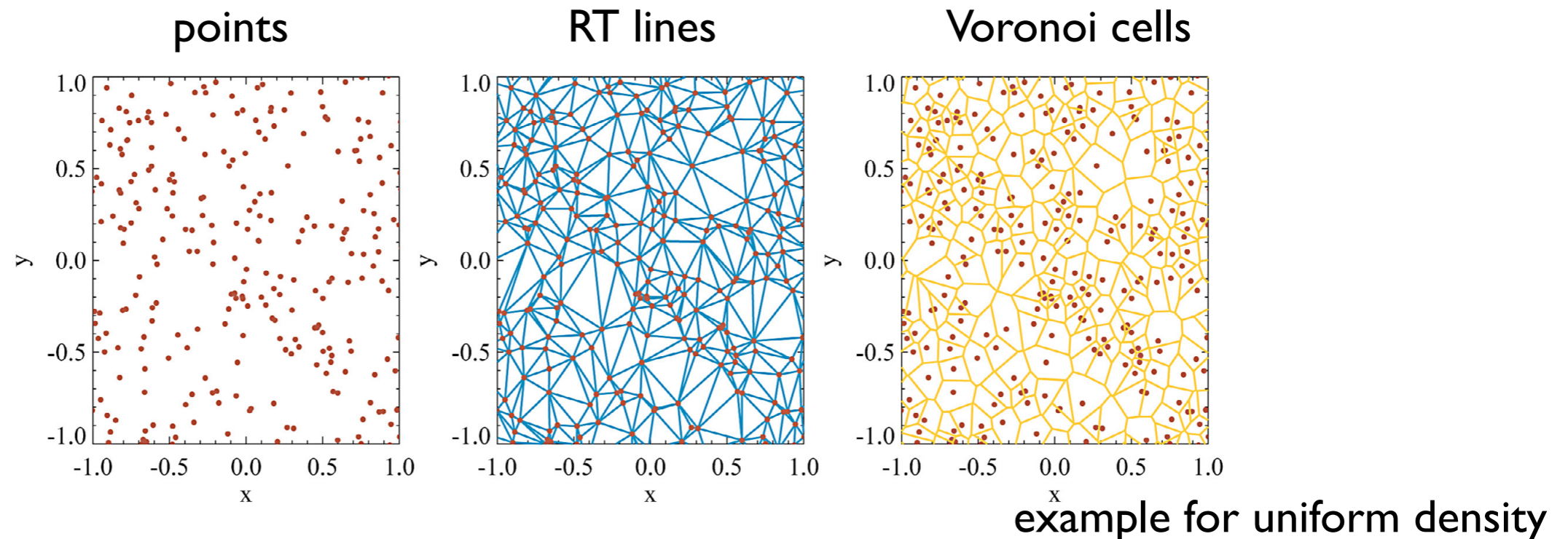
- *Only* follow RT from grid point to grid point
- Choose direction randomly with probability largest along continuation of incoming ray



- Based on SIMPLEX method (Ritzerveld & Icke 2006) developed for continuum radiative transfer
- Main difference: for line transfer, opacity changes from iteration to iteration, so separation no longer fixed in opacity.

# LIME

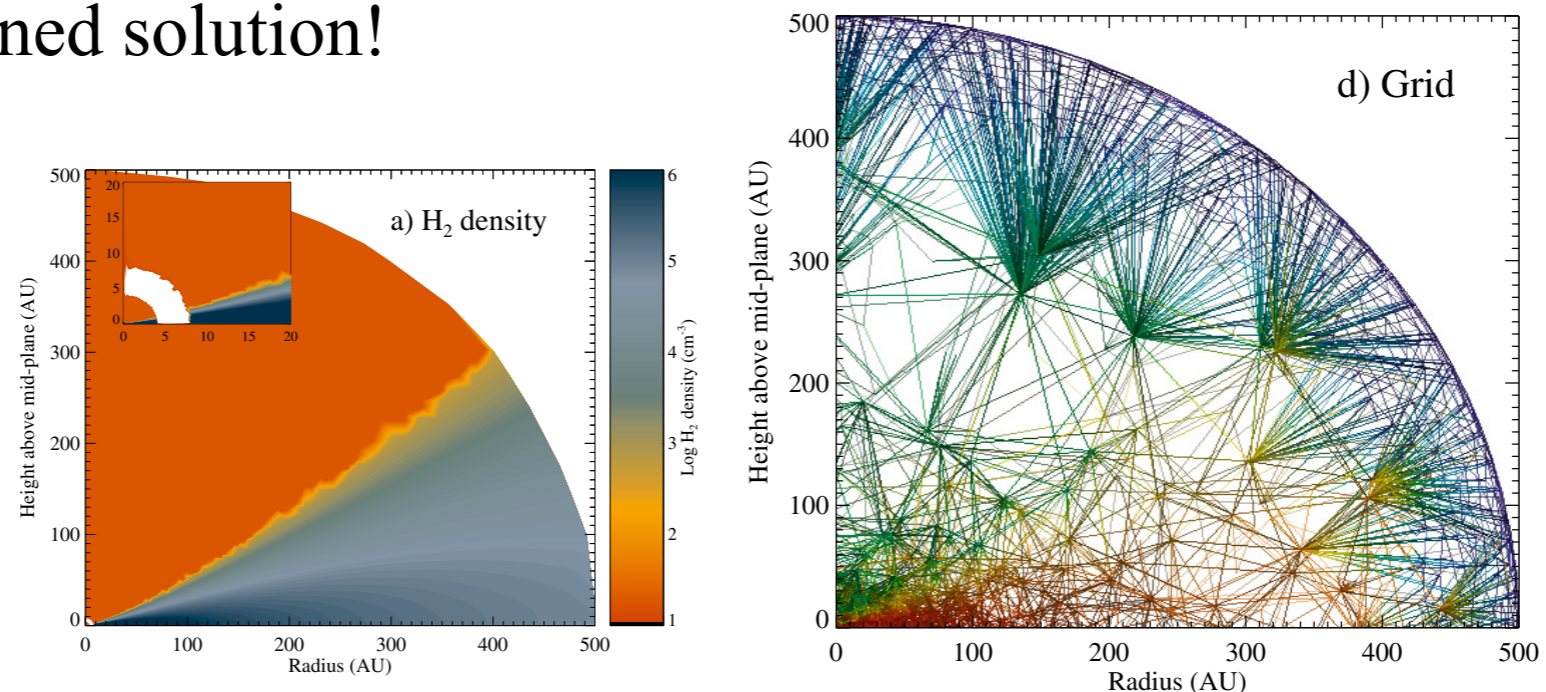
- *Only* follow RT from grid point to grid point
- Choose direction randomly with probability largest along continuation of incoming ray



- Properties of grid *point* are applied to all locations closer to that point than to any other grid point (Voronoi tessellation)

# LIME

- Some smoothing of grid points needed to avoid very small separation of points
- Need outer layer of grid points in empty space to connect photon paths to the rest of the Universe
- Very fast even for high opacities, can therefore use very many grid points
- One calculation for excitation and sky-image: cannot output populations for so many grid points
- As always: make sure afterwards that the grid is appropriate for the obtained solution!



# Outline

- How Monte Carlo methods can help
- The trouble with Monte Carlo methods
- By-passing some of these problems:  
accelerated methods
- Bernes, RATRAN, and LIME

➔ **How to work with LIME**

# Installing LIME

- <http://www.nbi.dk/~brinch/lime.php>

**LIME**  
Line Modeling Engine

home research publications thesis **LIME**

LIME is a new and innovative non-LTE spectral line radiation transfer code for 3D models in arbitrary geometries. The transport is done on a random density-weighted Delaunay grid. The code can predict line strengths and profiles of molecular transitions as well as the intensity of the thermal continuum radiation for disks around young stellar objects, protostellar envelopes, molecular clouds and similar environments.

Current version: 1.3, March 19, 2013  
LIME is now available for download from the **user forum**.

Documentation

- LIME cheat sheet
- LIME user manual
- LIME paper

**LIME links**

- LIME user forum
- Installation notes
- Molecular data files
- Dust opacities
- MIRIAD image analysis
- RADMC-3D
- RATRAN
- RADEX
- ARTIST

Register at the user forum to download LIME

# Installing LIME

- Here's how to install on a Mac
  - make sure `xcode` is up-to-date
  - get macports
  - `> sudo port install gsl`
  - `> sudo port install qhull`
  - `> sudo port install cfitsio`
  - download LIME
  - `> source sourceme.(csh|bash)`
  - `lime model.c`
  - download `paraview`

LIME requires qhull,  
gsl, and cfitsio

this installs LIME

this runs LIME for a  
model defined in  
model.c

use paraview to inspect  
the grid

# Running LIME

- Define your model
- Define weighting scheme (experts only!)
- Define molecule, transitions, distance, orientation
- `> lime model.c`
- **Output:** `grid.vtk`, `populations.pop`, `image0.fits`



# Example: a power-law sphere

- The ‘standard’ LIME example, calculating  $\text{HCO}^+ J=3-2$
- spherical cloud, 0.5–2000 AU
- $n_{\text{H}_2} = 1.5 \times 10^6 (r/300\text{AU})^{-1.5} \text{ cm}^{-3}$
- T 45...15 K (tabulated)
- $X(\text{HCO}^+) = 1 \times 10^{-9}$
- $b = 0.2 \text{ km/s}$
- $v$ : free fall to  $1.0 M_{\text{sun}}$  object
- $d = 140 \text{ pc}$

# Defining the LIME model

- Edit `model.c`

```

csh
/*
 * model.c
 * LIME, The versatile 3D line modeling tool
 *
 * Created by Christian Brinch on 11/05/07.
 * Copyright 2006-2013, Christian Brinch,
 * <brinch@nbi.dk>
 * Niels Bohr institutet
 * University of Copenhagen
 * All rights reserved.
 */

#include "lime.h"

/*****

void
input(inputPars *par, image *img){
/*
 * Basic parameters. See cheat sheet for details.
 */
par->radius          = 2000*AU;
par->minScale        = 0.5*AU;
par->pIntensity      = 4000;
par->sinkPoints      = 3000;
par->dust            = "jena_thin_e6.tab";
par->moldatfile[0]   = "hco+@xpol.dat";
par->antialias       = 4;
par->sampling        = 0;

par->lte_only        = 1;
par->outputfile      = "populations.pop";
par->binoutputfile   = "restart.pop";
par->gridfile        = "grid.vtk";

/*
 * Definitions for image #0. Add blocks for additional images.
 */
img[0].nchan        = 60;           // Number of channels
img[0].velres       = 500.;         // Channel resolution in m/s
img[0].trans        = 3;           // zero-indexed J quantum number
img[0].pxls         = 100;         // Pixels per dimension
img[0].imgres       = 0.1;         // Resolution in arc seconds
img[0].theta        = 0.0;         // 0: face-on, pi/2: edge-on
img[0].distance     = 140*PC;       // source distance in m
img[0].source_vel   = 0;           // source velocity in m/s
img[0].unit         = 0;           // 0:Kelvin 1:Jansky/pixel 2:SI 3:Lsun/pixel 4:ta\
u
img[0].filename     = "image0.fits"; // Output filename
----:--F1 model.c      Top L1      (C/1 Abbrev)-----
Loading cc-mode...done

```

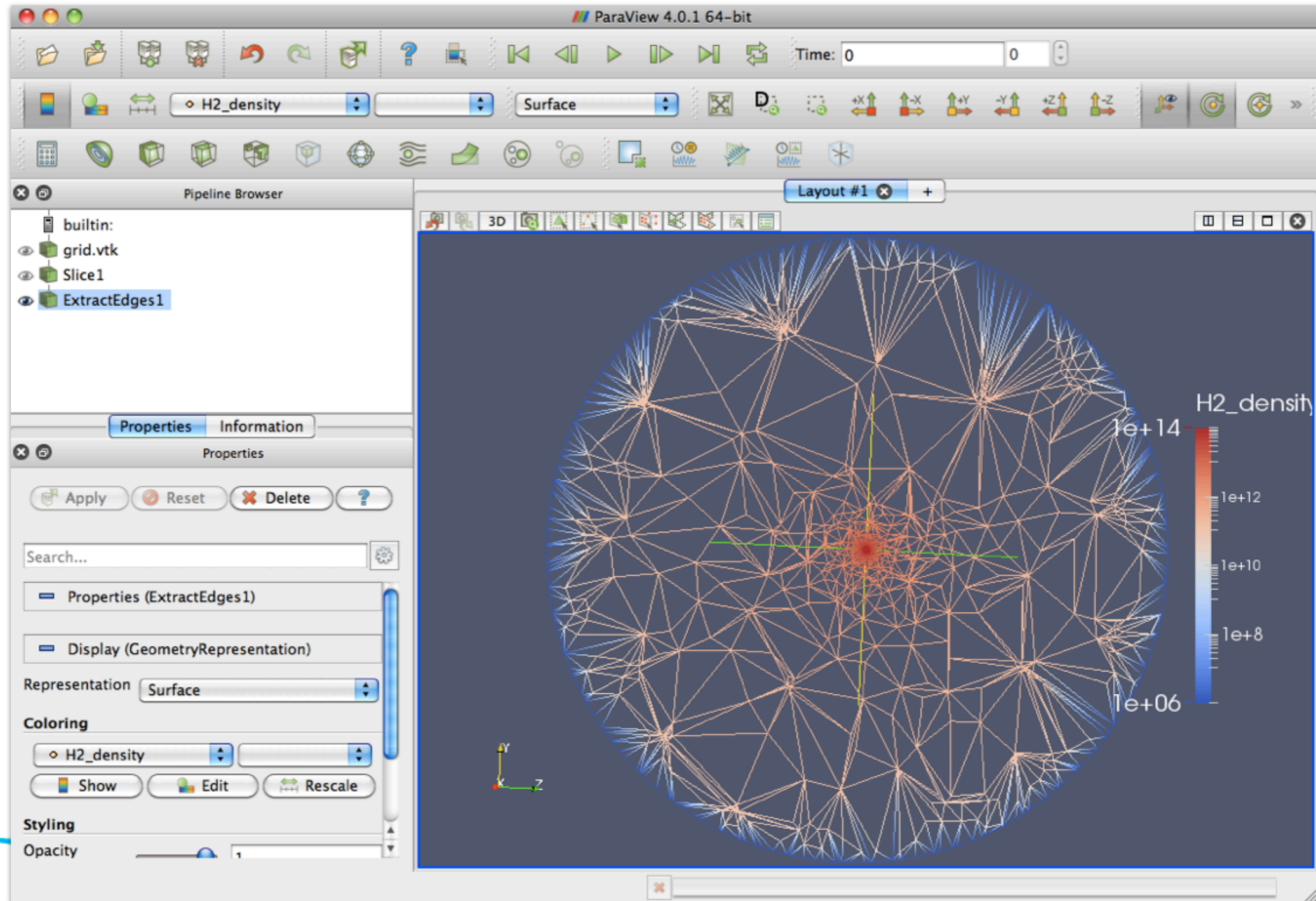
# Running LIME

- `> lime model.c`

```
*** LIME, The versatile line modeling engine, Ver.1.3
*** Copyright 2006--2013, Christian Brinch <brinch@nbi.dk>

Building grid      : ##### |           | Molecular data
Smoothing grid    : ##### |           |
Statistics        :           |           |
Iterations         :           |           |
Propagating photons:           |           |
Ray-tracing model :           |           |
```

# Inspecting the grid with paraview



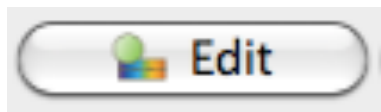
# paraview commands



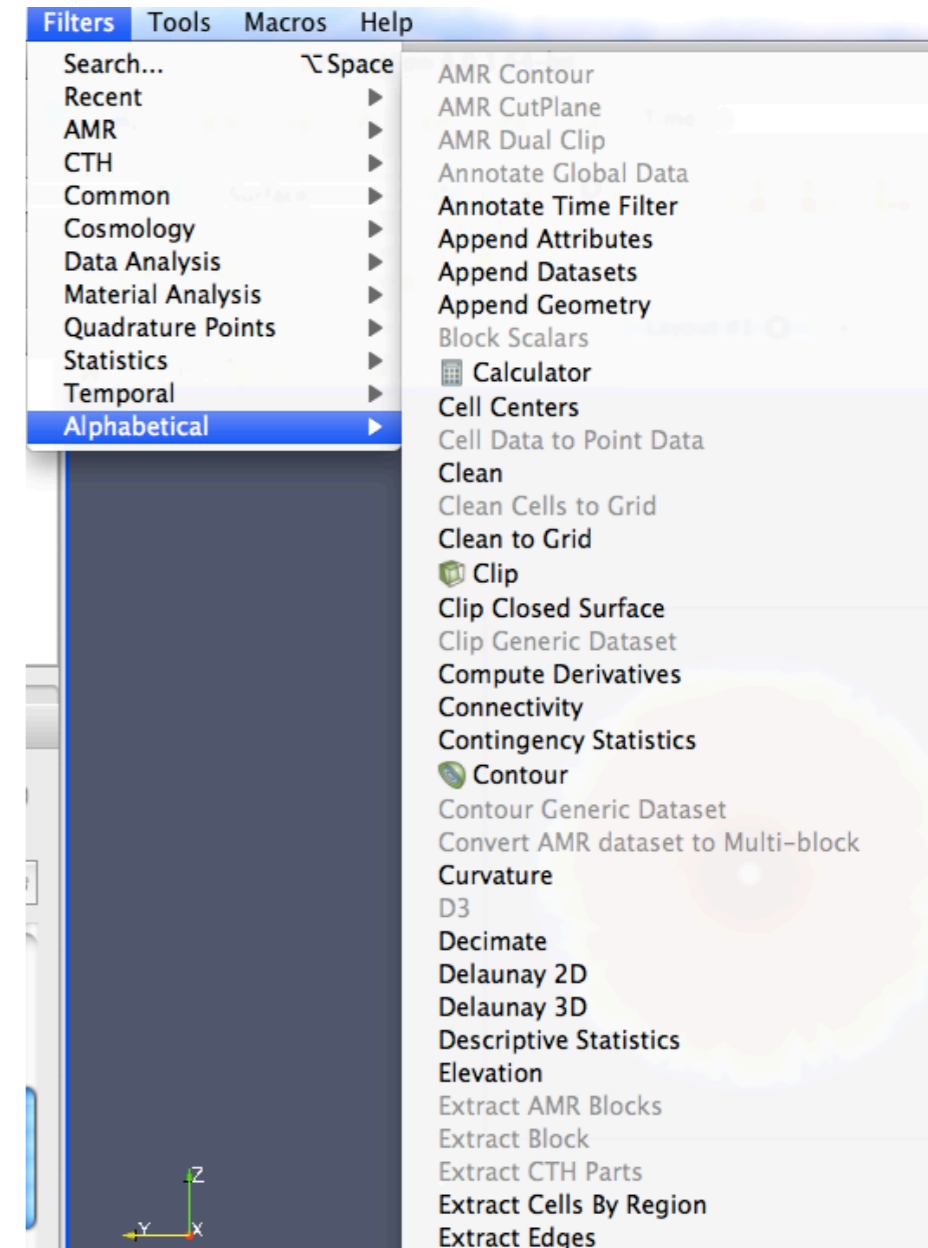
- open `grid.vtk` → apply



- select slice → apply and rotate



- edit color scheme → apply
- filters → alphabetical → extract edges → apply



# Monitoring progress

```
*** LIME, The versatile line modeling engine, Ver.1.3
*** Copyright 2006--2013, Christian Brinch <brinch@nbi.dk>

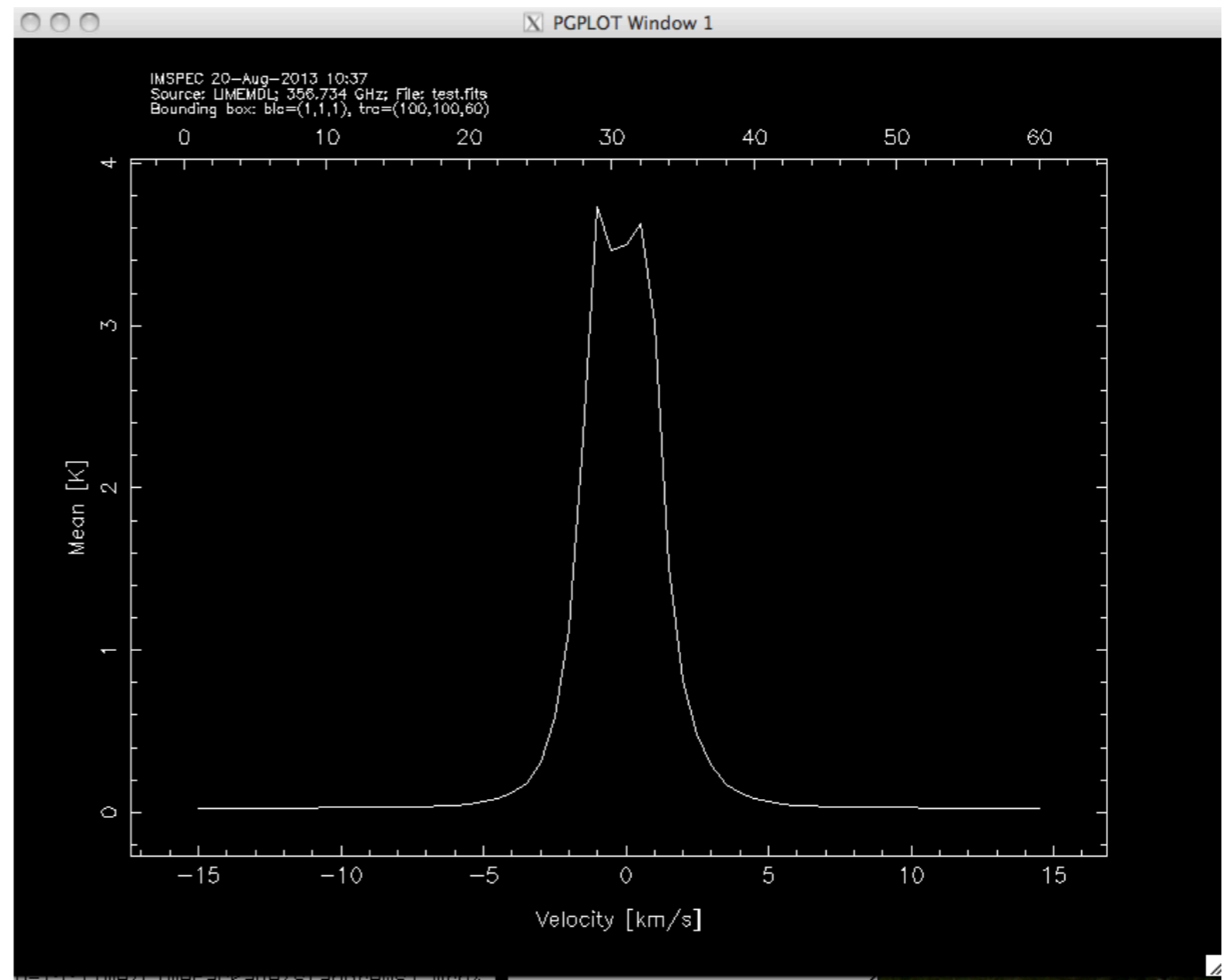
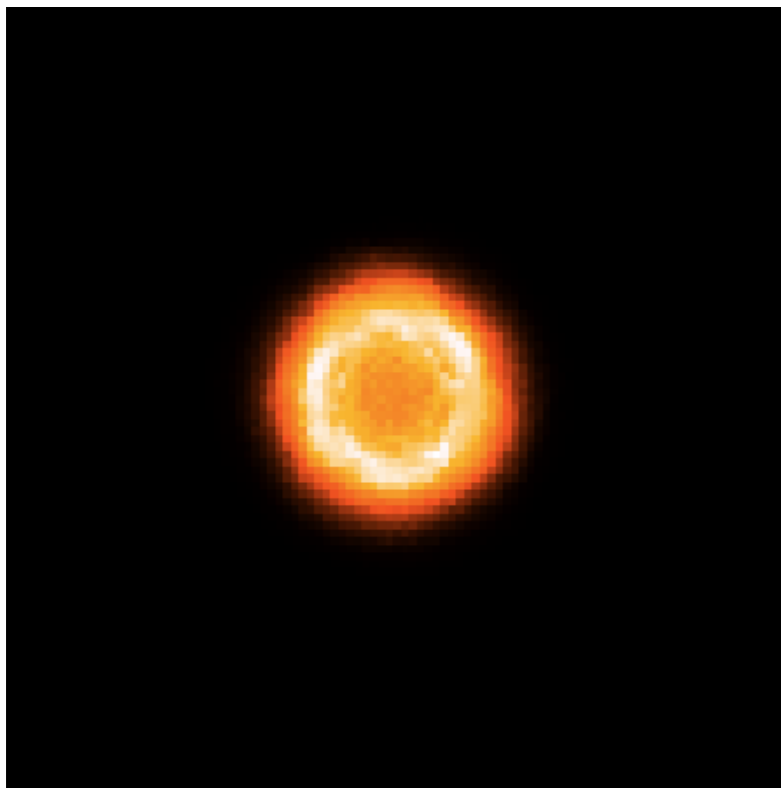
Building grid      : ##### |
Smoothing grid    : ##### | [ok]
Statistics        : Min(SNR)   3.316
                  : Median(SNR) 4.565
Iterations        : #####
Propagating photons: #####█
Ray-tracing model :

Molecular data
HCO+
1 collision partner(s):
H2

Model provides:
1 density profile(s)
```

# Inspecting the output

- for example using `ds9` or other programs to process submm data

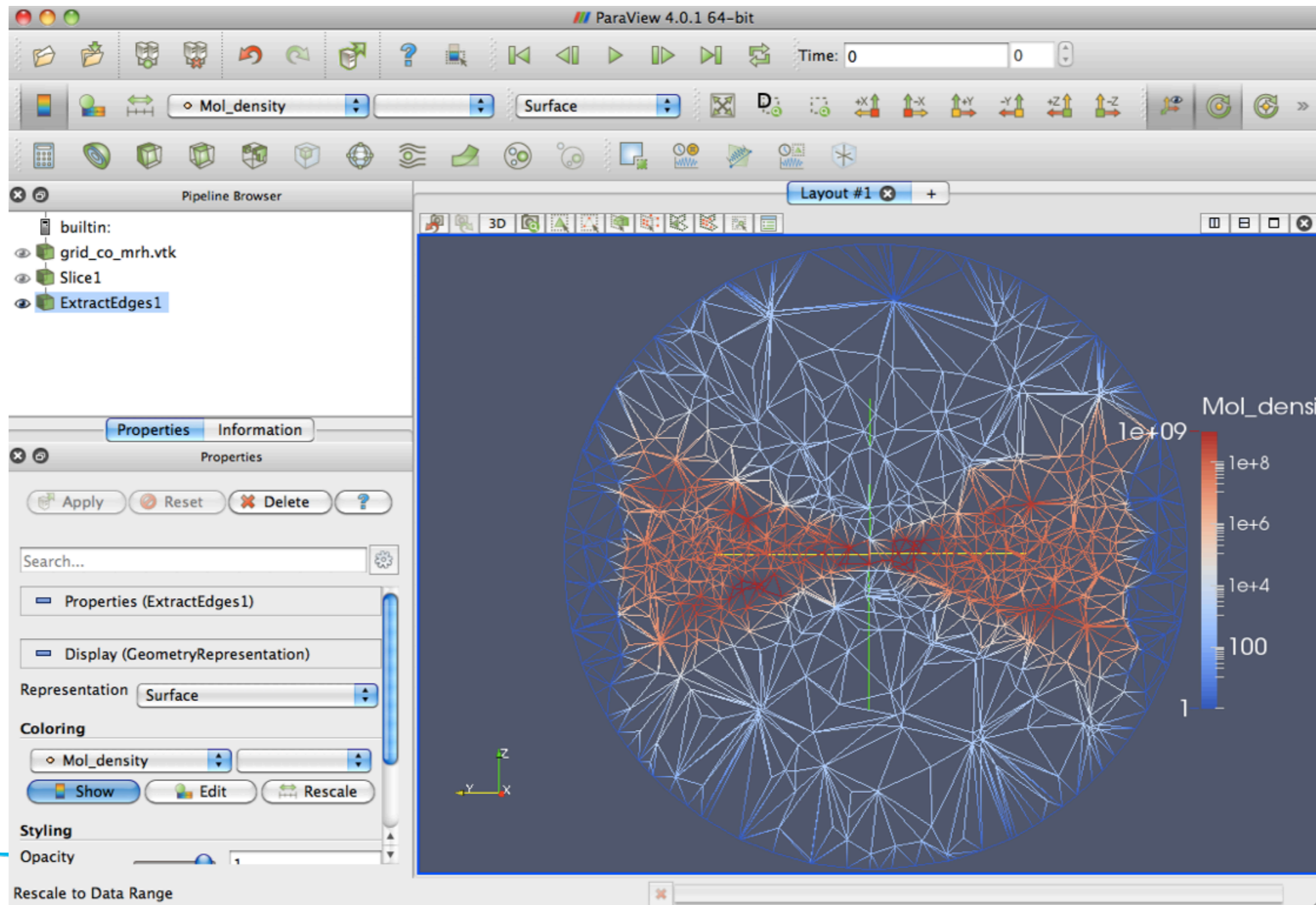


# Example: a circumstellar disk

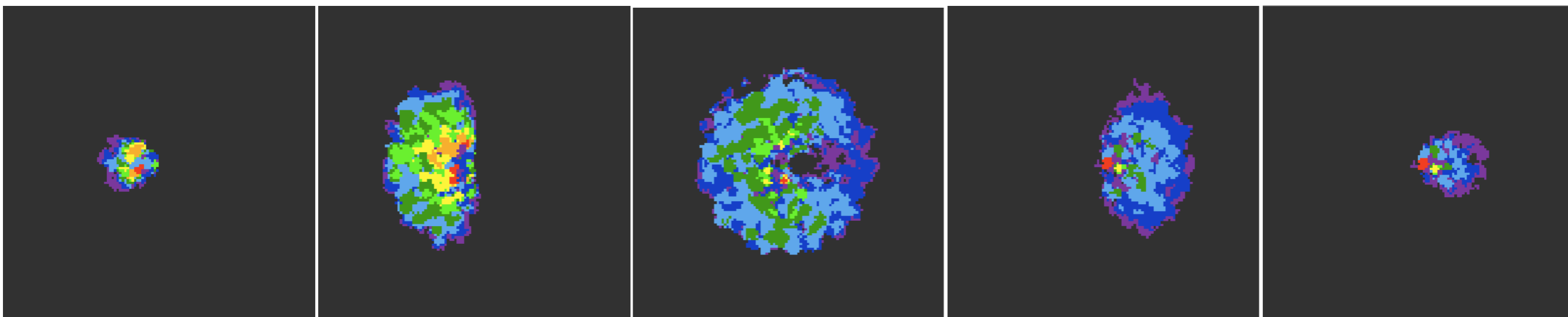
- disk mass  $0.01 M_{\text{sun}}$
- stellar mass  $0.6 M_{\text{sun}}$
- $R_{\text{out}}=250 \text{ AU}$
- $d=56 \text{ pc}, i=6^\circ$
- CO 3–2:  $X=2 \times 10^{-7}$  if  $T < 19\text{K}$ ,  $2 \times 10^{-4}$  if  $T > 19\text{K}$



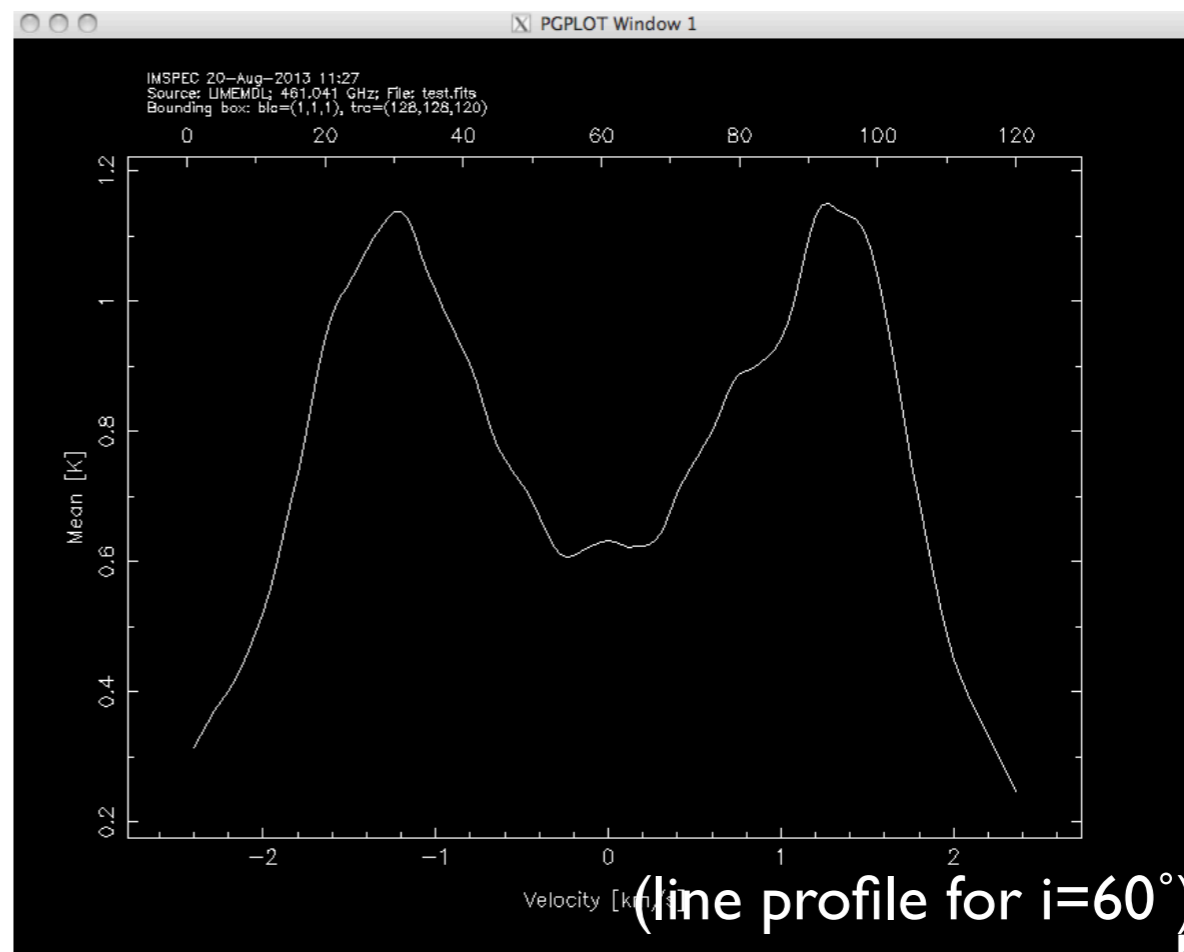
# Example: a circumstellar disk



# Example: a circumstellar disk



Channel maps and double-peaked line profile: Keplerian rotation



# Manual and Cheat Sheet

## Lime Cheat sheet

Version 1.3

Christian Brinch, 2013

```
void
input(inputPars *par, image *img)
{
    par->parameter = value;
    img[i].parameter = value;
}
```

### Required parameters:

(double) par->radius  
Outer model radius in meters.

(double) par->minScale  
Smallest scales sampled by grid.

(integer) par->pIntensity  
Number of grid points.

(integer) par->sinkPoints  
Number of surface grid points.

### Optional parameters:

(integer) par->sampling  
0 for spherical sampling, 1 for Cartesian sampling. Default is 0.

(double) par->tcmb  
Temperature of the microwave background. The default value is 2.725 K.

(string) par->moldatfile[i]  
Path to the i'th molecular data file.

(string) par->dust  
Path to dust opacity table. The moldatfile and dust parameters are optional in the sense that at least one of them (or both) should be set.

(string) par->outputfile  
Path to level population ascii output.

(string) par->binoutputfile  
Path to level population binary output used to restart LIME.

(string) par->restart  
File used to restart LIME with previously calculated populations stored in binary format.

(string) par->gridfile  
Path to VTK grid output file.

(string) par->pregrid  
Path to a file containing a predefined grid.

(integer) par->lte\_only  
Perform an LTE calculation only.

(integer) par->blend  
Set this parameter to take line blending into account. The default is unset.

(integer) par->antialias  
This parameter determines the level of antialiasing in the output images. A higher number makes better images but the code takes longer to run.

(integer) par->polarization  
For continuum polarization calculations, this parameter should be set to 1. Default is 0.

### Image parameters:

The img structure is an array so that img[i].value denotes the setting for the i'th image. Multiple images can be defined, i.e., img[0], img[1], etc.

### Required image parameters:

(integer) img[i].pxls  
Number of pixels per dimension.  
(double) img[i].imgres  
Image resolution in arcsec per pixel.

(double) img[i].theta  
Inclination from 0 (face-on) to  $\pi/2$  (edge-on)

(double) img[i].distance  
Object distance in meters.

(integer) img[i].unit  
0 for Kelvin, 1 for Jansky per pixel, 2 for SI units, and 3 for Solar luminosities per pixel. 4 gives a tau image cube.

(string) img[i].filename  
Path to the output fits image file.

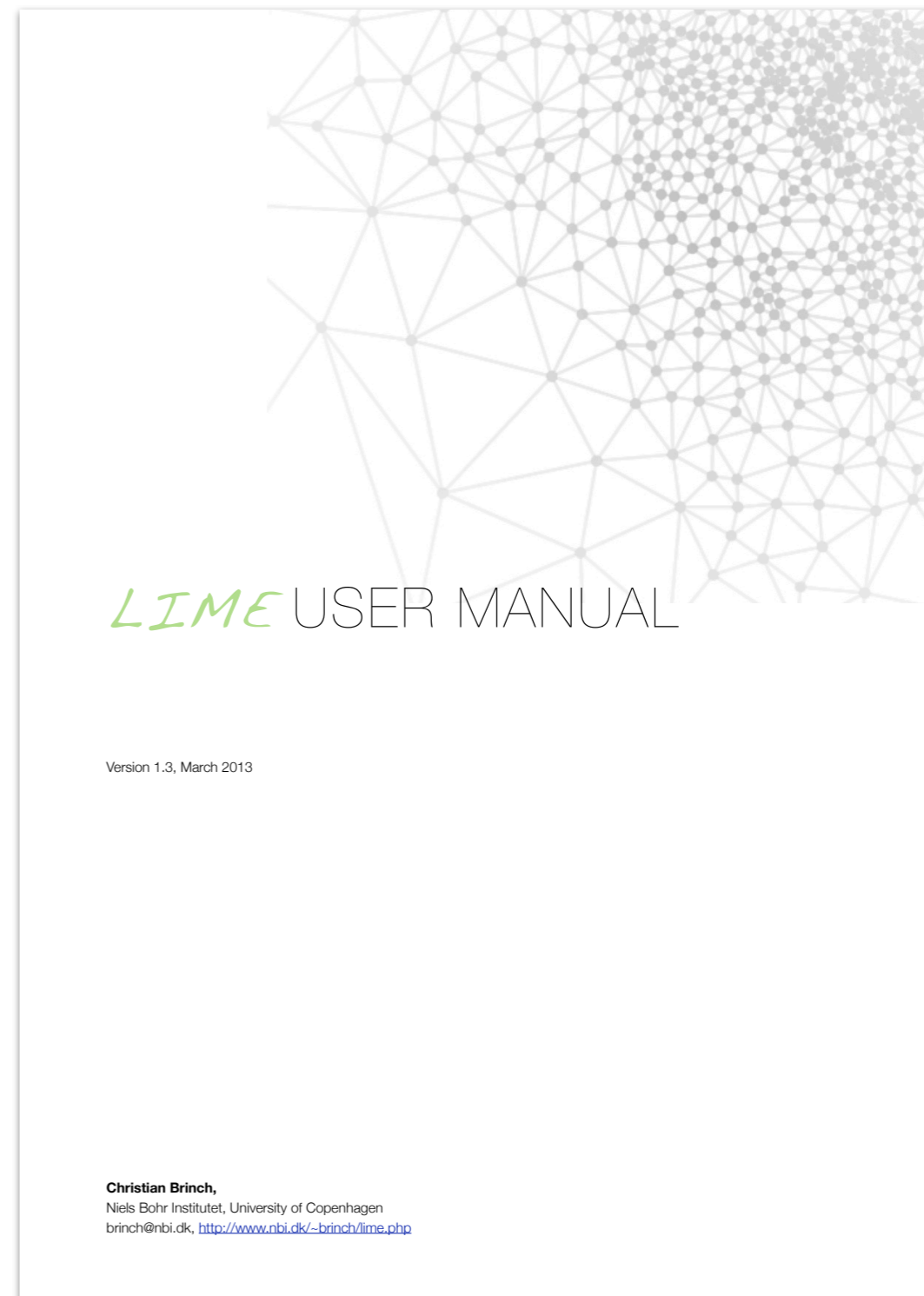
### Optional image parameters:

(double) img[i].phi  
Object rotation from 0 to  $2\pi$ . Default is 0.

(double) img[i].source\_vel  
Source velocity offset. Default is 0.

(integer) img[i].nchan  
Number of frequency channels.

(double) img[i].velres  
Velocity resolution in meters per second.



Version 1.3, March 2013

Christian Brinch,  
Niels Bohr Institutet, University of Copenhagen  
brinch@nbi.dk, <http://www.nbi.dk/~brinch/lime.php>

# A GUI interface: ARTIST

- ARTIST provides a GUI for running LIME
- use pre-defined star-formation models *or* a user-defined input model
- less flexibility and less customization than stand-alone LIME, but great for the novice user
- <http://youngstars.nbi.dk/artist/Welcome.html>

# Further reading

- van Dishoeck & Blake 1998, ARA&A 36, 317
- Tielens: The Interstellar Medium (book)
- Rybicki & Lightman: Radiative Processes in Astrophysics (book)
  
- RADEX: [www.strw.leidenuniv.nl/~moldata/radex.html](http://www.strw.leidenuniv.nl/~moldata/radex.html)
  - <http://www.strw.leidenuniv.nl/~brinch/website/lime.html>
  
- LAMDA: [www.strw.leidenuniv.nl/~moldata](http://www.strw.leidenuniv.nl/~moldata)
  - <http://www.strw.leidenuniv.nl/~brinch/website/lime.html>
  
- RATRAN: [www.strw.leidenuniv.nl/~michiel](http://www.strw.leidenuniv.nl/~michiel)
  - <http://www.strw.leidenuniv.nl/~brinch/website/lime.html>
  
- LIME: <http://www.strw.leidenuniv.nl/~brinch/website/lime.html>
  - <http://www.strw.leidenuniv.nl/~brinch/website/lime.html>