SimLine - Radiative transfer in molecular lines

Version 2.12 (4/11/2012)

Dr. V. Ossenkopf

I. Physikalisches Institut der Universität zu Köln

Users Manual

Abstract

SimLine (previously called LTR) is a FORTRAN code to compute the profiles of molecular rotational lines in spherically symmetric clouds with arbitrary density, temperature and velocity structure. Turbulence and clumping effects are treated in a local statistical approximation combined with a radial dependence of the correlation parameters. The code consists of two parts: the self-consistent solution of the balance equations for all level populations and energy densities at all radial points and the computation of the emergent line profiles observed from a telescope with finite beam width and arbitrary offset. All numerical discretizations are done in an adaptive way, the user has full error control and the line profiles may be computed in almost arbitrary accuracy. An accelerated lambda iteration is used to solve the equation system. The optical depths in the lines may vary between about -5 and +5000. The input of physical cloud parameters, telescope parameters, and numerical control parameters may be done either interactively or as file input. Molecular data are read from table files.

1 The basic theory

1.1 The solution of the radiative transfer problem

The general formulation of the frequency dependent radiative transfer equation

$$\frac{dI(\nu,\vec{r},\vec{n})}{ds} = -\kappa(\nu,\vec{r},\vec{n})I(\nu,\vec{r},\vec{n}) + \epsilon(\nu,\vec{r},\vec{n}) \tag{1}$$

may be reduced in the spherically symmetric case down to three independent coordinates. It is sufficient to consider radiation in the direction of the z axis and to neglect the ϕ dependence of the radius vector. Then, the radiative transfer equation can be written as

$$\frac{dI_z(\nu,p,z)}{dz} = -\kappa(\nu,p,z)I_z(\nu,p,z) + \epsilon(\nu,p,z) \tag{2}$$

where p denotes the displacement variable ($p = \sqrt{x^2+y^2}$). The z axis is chosen in the direction towards the observer. Since the spatial grid is adjusted at distinct radial points, the program does not use the form of the radiative transfer equation with source function and derivative on the optical depth, but with emission and absorption coefficients.

For the numerical solution of the radiative transfer equation it is convenient to use stepwise the solution of the integral equation allowing a reduction of the error to third order

$$I_z(\nu,p,z_i) = \exp\left(-\int_{z_{i-1}}^{z_i}\kappa(\nu,p,z)dz\right)\left[I_z(\nu,p,z_{i-1}) + \int_{z_{i-1}}^{z_i}\epsilon(\nu,p,z)\exp\left(\int_{z_{i-1}}^{z}\kappa(\nu,p,z')dz'\right)dz\right] \tag{3}$$

The incident radiation at the outer boundary of the cloud is assumed to follow a black body spectrum with temperature T_{bg}.

$$I_{\rm bg}(\nu) = \frac{2h\nu^3}{c^2}\left[\exp\left(\frac{h\nu}{k\times T_{bg}}\right)-1\right]^{-1} \tag{4}$$

For the cosmic background radiation $T_{bg} = 2.73K$.

The program assumes complete redistribution of energy between the molecules at a given level, so that one can define a unique set of level populations n_j at any point $\vec{r}$ which is independent from frequency or the actual velocity of a molecule. Furthermore, this means that the line profiles for emission and absorption are the same. In molecular clouds it is sufficient to consider pure Doppler broadening so that the profiles are given by

$$\Phi(\nu) = \frac{1}{\sqrt{\pi}\sigma}\exp\left(-\frac{(\nu-\nu_0(1+v_z/c))^2}{\sigma^2}\right) \tag{5}$$

where v_z is the systematic velocity in z direction, σ is the local line width, and ν is the frequency in the observers frame. (Without loss of generality, the program assumes that the centre of the cloud is in rest within the observers frame.).

With the assumtion of complete redistribution, the absorption and emission coefficients for a given transition are determined by

$$\kappa_{j,l}(\nu) = \frac{h\nu_0}{c}\left(n_l B_{l,j} - n_j B_{j,l}\right)\Phi(\nu) \tag{6}$$

$$\epsilon_{j,l}(\nu) = \frac{h\nu_0}{4\pi}\, n_j A_{j,l}\Phi(\nu) \tag{7}$$

where stimulated emission is treated as negative absorption.

The level populations at a given radius r may be computed from the linear system of balance equations

$$n_j\sum_{l\neq j}\left(A_{j,l} + B_{j,l}u_{j,l} + c_{j,l}\right) = \sum_{l\neq j} n_l\left(A_{l,j} + B_{l,j}u_{j,l} + c_{l,j}\right) \tag{8}$$

where the radiative energy density $u_{j,l}$ for the transition between the levels j and l at a given point is computed from the integral

$$u_{j,l}(p,z) = \frac{1}{c}\int_0^{2\pi}d\phi\int_0^{\pi}d\theta\,\sin\theta\int_{-\infty}^{\infty}d\nu\,\Phi_{\phi,\theta}(\nu,\vec{v})I(\nu,p,z,\phi,\theta)\ . \tag{9}$$

Exploiting the spherical symmetry, this may be transformed into

$$u_{j,l}(r) = \frac{2\pi}{rc} \int_{-r}^{r} dz' \int_{-\infty}^{\infty} \Phi(\nu) I_z(\nu, p' = \sqrt{r^2 - z'^2}, z') d\nu \tag{10}$$

where the radius r is given as $r = \sqrt{p^2 + z^2}$. Due to the symmetry, only radiation parallel to the z axis is encountered. For the central line frequency ν_0 within the expression for Φ (Eq. 5), the frequency of the transition $j \to l$ has to be taken when computing $u_{j,l}$.

The equation system is truncated whenever the excitation of a level falls below the chosen accuracy limit or at the maximum level for which the collision rates are known.

1.2 The local turbulence approximation

The turbulence description uses two parameter for each point $\vec{r}$: the total width of the velocity distribution σ giving the local profile for optically thin lines and the correlation length $r_{\rm corr}$.

The width of the velocity distribution σ is composed of a turbulent and a thermal contribution

$$\sigma = \frac{\nu_0}{c} \sqrt{\frac{2kT_{\rm kin}}{m} + \frac{2}{3}\langle v_{\rm turb}^2 \rangle} \tag{11}$$

where a Maxwellian distribution of turbulent velocities is assumed. In the data input, the turbulent velocities are given as FWHM of the velocity distribution (FWHM($v_{\rm turb}$) = $\sqrt{8/3 \times \ln 2 \, \langle v_{\rm turb}^2 \rangle}$). The thermal velocities are automatically computed from the kinetic gas temperature.

The long range correlation of the turbulence spectrum as described by a Kolmogorov exponent can be simulated in 1D by a radial dependence of the turbulent velocity dispersion $\langle v_{\rm turb}^2 \rangle \propto r^\gamma$. Observations of Fuller & Myers (1992) suggest exponents around $\gamma \approx 0.7$. Several other authors proposed somewhat smaller values around $\gamma \approx 0.5$.

For the local treatment of clumping in real space or velocity space the considered volume element is subdivided into numerous clumps whith a thermal internal velocity dispersion. From a Gaussian distribution of the density of molecules with a certain velocity within the single fragments

$$n(r) = n_0 \times \exp(-r^2/r_{\rm cl}^2) \tag{12}$$

we get an effective absorption coefficient for the whole medium given by

$$\kappa_{\rm eff} = n_{\rm cl} \times \pi r_{\rm cl}^2 \int_0^{\tau_{\rm cl}} \frac{1 - \exp(-\tau)}{\tau} d\tau \tag{13}$$

(Martin et al. 1984), where $n_{\rm cl}$ is the number density of clumps which contribute at the considered frequency and $\tau_{\rm cl} = \sqrt{\pi}\, \kappa r_{\rm cl}$ is their central opacity.

Assuming a Maxwellian turbulent velocity distribution of clumps and a thermal velocity dispersion $\sigma_{\rm th}$ giving at most 1/3 of the total velocity dispersion σ, we obtain an effective absorption coefficient

$$\kappa_{\rm eff}(\nu) = n_{\rm ges} \pi r_{\rm cl}^2 \times A(\tau_{\rm cl}) \times \frac{\sigma_{\rm th}}{\sigma} \exp\left(-\frac{(\nu - \nu_0)^2}{\sigma^2}\right) \tag{14}$$

with

$$A(\tau) = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} dv \int_0^{\tau \exp(-v^2)} \frac{1 - \exp(-\tau')}{\tau'} d\tau' \tag{15}$$

Here, $n_{\rm ges}$ is the total number of clumps. In case of pure turbulence in velocity space, $n_{\rm ges}$ is given by the volume of the single clumps and we obtain

$$\kappa_{\rm eff}(\nu) = \kappa(\nu) \times A(\tau_{\rm cl})/\tau_{\rm cl} \; . \tag{16}$$

Consequently, $A(\tau_{\rm cl})/\tau_{\rm cl}$ is a measure of the reduction of the opacity due to turbulence. For small clump sizes, $A(\tau_{\rm cl}) = \tau_{\rm cl}$ so that we are in the microturbulent limit. For $\tau_{\rm cl} \gg 1$ the macroturbulent limit is reached. In case of real clumping in space, $\kappa_{\rm eff}(\nu)$ is further reduced by the filling factor. In the program, this is simulated by a corresponding artificial reduction of the molecular abundance.

The clumps size $r_{\rm cl}$ is determined by the correlation length of the velocity or density structure. $r_{\rm cl}$ is the length on which the abundance of molecules within the same thermal velocity profile is

reduced by the factor 1/e. Consequently, it can be computed from the turbulence correlation length by $r_{\rm cl} = r_{\rm corr} \times \sigma_{\rm th}/\sigma$.

One has to keep in mind that this is a statistical approximation, always assuming a complete Maxwellian velocity distribution within each volume element even if the single clumps become relatively large. Furthermore, the source function $S = \epsilon(\nu)/\kappa(\nu)$ is not influenced in this local approach.

1.3 The central H II region

For the simulation of the internal heating produced by a central continuum source in the cloud, it is possible to assume an H II region in the cloud core. The H II region is characterized by two parameters, the electron density $N_{\rm e}$ and the kinetic electron temperature $T_{\rm e}$.

The absorption coefficient for electron-ion bremsstrahlung in the Rayleigh-Jeans approximation is given by:

$$\kappa(\nu) = \frac{8}{3\sqrt{2\pi}} \frac{e^6}{(4\pi\epsilon_0 m_{\rm e})^3 c} \left(\frac{N_{\rm e}}{\nu}\right)^2 \left(\frac{m_{\rm e}}{kT_{\rm e}}\right)^{3/2} \ln\Lambda \tag{17}$$

where it is assumed that the gas is singly ionized and Λ is given by

$$\Lambda = \left(\frac{2kT_{\rm e}}{\delta m_{\rm e}}\right)^{3/2} \frac{4\pi\epsilon_0 m}{\pi\delta e^2 \nu} \approx 4.9573\,10^7 \left(\frac{T}{\rm K}\right)^{3/2} \frac{\rm Hz}{\nu} \tag{18}$$

for $T_{\rm e} < 3.2\,10^5$K. The quantities e and $m_{\rm e}$ denote the electron charge and mass and c is the vacuum light velocity.

For a thermal plasma, the emission coefficient follows from the Planck function

$$\epsilon(\nu) = \kappa(\nu) \times B_\nu(T_{\rm e})\ . \tag{19}$$

In the radiative transfer computations within the H II region the small frequency dependence of these continuum coefficients within the molecular rotational lines is neglected.

1.4 The Sobolev approximation

As an initial guess to the level populations, the Sobolev approximation may be used. This approximation is strictly valid for clouds with large velocity gradients and velocities which are increasing outwards so that all radiative coupling is restricted to a small spatial region. Then the level populations can be computed from local quantities only.

The radiative energy density at a radius r is determined by the local source function $S_{j,l}(r) = \epsilon_{j,l}(r)/\kappa_{j,l}(r)$

$$u_{j,l}(r) = (1 - \beta_{j,l}(r))\, S_{j,l}(r) + \beta_{j,l}(r) I_{\rm bg}(\nu_0) \tag{20}$$

where $\beta_{j,l}(r)$ denotes the probability for the escape of a photon within the line j from the cloud when emitted at r. It can be computed from an integral over all spatial directions from a given point by

$$\beta_{j,l}(r) = \frac{1}{2} \int_{-1}^{1} \frac{1 - \exp(-\kappa_{j,l}(r)/Q(r,\mu))}{\kappa_{j,l}(r)/Q(r,\mu)}\, d\mu \tag{21}$$

with the line integrated effective absorption coefficient $\kappa_{j,l}$ and the velocity gradient in spherical symmetry

$$Q(r,\mu) = \mu^2 \frac{dv_r}{dr} + (1-\mu^2)\frac{v_r}{r} \tag{22}$$

Hence, the radial velocity gradient is the most important quantity for the local energy density.

In contrast to the iterative solution of Eq. (8) it is possible to apply a more efficient way here, since the energy densities are analytically coupled to the level populations via Eq. (7) and (20-22). Using the derivatives of the matrix coefficients, the balance equations can be solved by a Newton-Raphson approach. The corrections to the level populations within each step will be computed from the matrix equation

$$\sum_i \left(\sum_k \frac{\partial A_{jk}}{\partial n_i} n_k + A_{ji} \right) \Delta n_i = \sum_i A_{ji} n_i \tag{23}$$

where the A_{ij} are the coefficients of the matrix of balance equations (Eq. 8).

1.5 Computation of beam temperatures

When the level populations are known, the beam temperature relative to the background is computed from the emergent intensity by integration over the projection of the telescope beam on the cloud.

$$T_{\rm beam}(\nu) = \frac{c^2}{2k\nu_0^2} \frac{\displaystyle\int_0^{2\pi} d\phi \int_0^{\infty} dp\, p \times (I_{\rm surf}(\nu,p) - I_{\rm bg}(\nu)) f_{\rm beam}(p,\phi)}{\displaystyle\int_0^{2\pi} d\phi \int_0^{\infty} dp\, p \times f_{\rm beam}(p,\phi)} \tag{24}$$

The emergent intensity is the value on the cloud surface $I_{\rm surf}(\nu,p) = I_z(\nu,p,\sqrt{R_{\rm cloud}^2 - p^2})$. A Gaussian profile for the beam is assumed

$$f_{\rm beam}(p,\phi) = \exp\left(\frac{-(p-p_{\rm offset})^2(1+\phi^2)^2}{\sigma_{\rm beam}^2}\right) \tag{25}$$

The projected beam width is computed from the angular width by $\sigma_{\rm beam} = \pi D^2/648000\ \sigma_{\rm beam}['']$ where D is the distance of the cloud. The standard deviation $\sigma_{\rm beam}$ is coupled to the full width of half power by FWHM $= 2\sqrt{\ln 2}\,\sigma_{\rm beam}$. The program computes a full map, i.e. the line profiles at a given number of positions on a linear radial scan through the cloud.

Within the program, all frequencies are treated in units of $(\nu-\nu_0)/\nu_0$, velocities in units of c, and intensities in units of ν_0^3/c^2, i.e. by a factor $4\pi/h$ larger than the SI unit.

2 The general code design

The general design of the program is directed towards a high accuracy of the computed line profiles. All of the errors in the different steps of the program are explicitely user controlled by setting thresholds. All discretizations necessary to treat the problem numerically are performed in an adaptive way, i.e. there is no predefined grid at all and all grid parameters may change during the iteration procedure.

Furthermore, the code was pushed towards a high flexibility, i.e. the ability to treat a very broad range of all physical parameters with the same accuracy and without numerical limitations. E.g., the systematic velocities may range from 0 to several times the turbulent velocity depending on the compiled frequency field size. There is no inherent restriction to a distinct range of optical depths. I have tested it for depths at line centre between about -5, i.e. moderate masering and about 5000. However, the convergence speed depends dramatically on the optical depths, the level structure and the local velocity gradients.

The program is not optimized towards a high velocity of the code. Although I got a factor of about 60 in velocity relative to the precursor program by E. Krügel, other codes with lower inherent accuracy may run another factor 10 faster. Nevertheless, the code is suitable for an interactive work even on a small PC.

3 User interface

3.1 Programm invocation

The radiative transfer code is started from the command line by:

`ltr` [options]

where options can be:

`-fileinput`

`-numdefaults`

`-overwrite`

`-fits`, `-block` or `-lines`

`-nocomments`, `-unixcomments` or `-vmscomments`

When the option `-fileinput` is given, the program assumes that all physical, observational and numerical parameters will be read from files; interactive parameter input is switched off. The option `-numdefaults` sticks the set of the used numerical parameters to the default values. The other options set the behaviour for the data file output (see Sect. 3.3).

All the options are intended to avoid additional interactive questions during the program execution. If options are omitted, the corresponding parameters will be determined interactively during the program execution.

3.2 Terminal interaction

The program is designed for an interactive computation of line profiles so that the flow is controlled by terminal input. However, all parameters necessary for the computation may be given either interactively or may be read from data files except for the molecular constants which must be provided in a file.

There is a strict separation between parameters describing the molecule, parameters for the cloud geometry and physics, parameters of the observation, and numerical parameters which control the accuracy of the different parts of the computation.

A parameter input which was done interactively may be stored immediately in a file with the appropriate structure. This file can be read at the next time of use, and modifications of single values can be done by means of a text editor. In case of errors made when interactively defining the data, the input procedure can be repeated by typing a character which cannot be converted to a numerical value at the next input question.

The two parts of the program may be run in arbitrary order and repeatedly. At first one can compute either a new cloud model, i.e. the level populations at all radial points throughout the cloud, or one can read a file which contains the results of a previous computation. Then, it is possible to compute the observed line profiles for an arbitrary number of lines or telescope parameters. For a hand-made line fit one can change some cloud parameters and repeat the whole procedure.

I recommend to use a graphics software in a parallel session/window to get an immediate impression from the changes on the line profiles.

3.3 Data files

3.3.1 Input files

All input files are plain ASCII files so that they can modified with any text editor. Output is either plain ASCII or FITS to enable easy examination, modification and use for further reduction in other programs.

The input files providing the molecular, physical, observational and numerical parameters consist of sets of dual lines. The first line in each pair is treated as a comment line by the program and, in general, contains a description of the parameter. The second line contains the parameter in ASCII form. Exceptions occur in the file for the molecular data where the levels, transitions and collisional rates are given in tables with only one comment line per table and in the cloud parameter file if the functions for the different quantities are given by a data table which has also only one comment line. I highly recommend to produce a first set of input files using the SimLine program itself by interactive input of the parameters. This guarantees the consistency of the produced parameter files with the structure expected when reading the parameters. Starting from this first set it may be more feasible to change distinct values in these files using a text editor.

3.3.2 Molecular data files

Molecular data are read from ASCII files with a simple table structure similar to but not compatible with the RADEX format. Data files for several molecules accompany the SimLine executables and should be used as general examples.

The file starts with a comment line describing the molecule and ends with an arbitrary number of comment lines for references. The entries within the file are given as single values or tables each preceeded by a comment line explaining it. The following entries occur

`Relative molecule mass:`
relative mass of the isotope given in atomic mass units

`Number of levels:`
At most 50 levels can be read.

`Number of transitions:`
At most 200 radiative transitions can be read.

`Levels: index, statistical weight, energy [cm^-1], name`
Here, a table with one line for each level has to be provided. Index numbering should start by 1. To denote each level a string of 8 characters may be used.

`Transitions: from index, to index, Einstein-As [s^-1]`
Here, a table with one line for each radiative transition has to be provided. The indices have to correspond to the values in the level table.

`Collision rates`
`Number of temperatures`
To separate the collision rates from the molecular radiation data two comment lines are introduced here. Collision rate coefficients can be tabulted for at most 10 different temperatures.

`Table of temperature values`
Here, the temperatures have to be given - one per line.

`Number of transitions`
At most 1000 collisional transitions can be read.

`Transitions from - to index, rate coefficient`
Here, a table with one line for each transition has to be provided. In each line the indices of the two levels and the rate coefficients for all temperatures have to be given.

`Comments and references`
At the end of the file an aritrary number of comment lines can be used where references to all data should be provided.

3.3.3 Output files

Two types of result files are produced. The first one contains the level populations calculated as the result of the lambda iteration. This file is only intended for further processing by SimLine in the computation of line profiles. There are no comments in the file. The first line contains the file name of the molecular data file. In the second line we find the number of radial points, the number of rotational levels treated and the background radiation temperature at the location of the cloud. In the following block, the physical parameters at the different radial points are stored. Each line contains the number of the radial point, its radius, the molecule density, the systematic velocity, the thermal+turbulent velocity σ, and the correlation length over σ. A second block contains the lines with the level populations. For each radial point (denoted by its number) the populations at all computed levels are stored. The same file structure as produced here must be given when reading a precomputed cloud model from a file in order to calculate new line profiles.

The second file type contains the line profiles. These files are designed to be readable by graphics programs as IDL, GREG or GNUPLOT. Two general file structures are possible - FITS and ASCII. In case of ASCII files one can further select whether to write the data points as a compact table or each point with offset and velocity in a separate line. In FITS format the lines are stored in a two-dimensional array with the first axis given by the angular offest and the second axis by the velocity. In ASCII block mode the two dimensional field of beam temperatures is stored as a block where the lines represent the different beam offsets and the columns the frequency/velocity points. In two preceding comment lines the offset and frequency values are given. The line-oriented structure requires much more disk space but can be read by more programs. Here, each line contains three numbers - the beam offset, the frequency, and the beam temperature. Consequently the file contains (number of offsets)$\times$(number of frequency points) lines. Preceeding comment lines specify these numbers. The format of comment lines can be chosen to be in VMS or UNIX format, i.e. whether a '!' or '#' shall be used to indicate the beginning of a comment line. Furthermore one can also skip the comment lines.

The default parameters for the structure of the line profile files and the possibility to always overwrite existing files with newer ones can be specified as options on the command line. The options `-fits`, `-block` and `-lines` turn on FITS format, ASCII block mode or ASCII line mode for the file structure. The options `-nocomments`, `-unixcomments`, and `-vmscomments` define the format of the comment lines in case of ASCII files. When the additional option `-overwrite` is given, existing files will be replaced without further notice. Otherwise, every overwrite has to be confirmed interactively. All these options can be given also on compile time via conditional defines (e.g. `-Dnocomments -Doverwrite`). If no parameters are specified, they will be asked interactively before each file output.

The units in the files are always pc for radii and km/s for velocities/frequencies. Beam offsets are given in degrees on FITS output and in arcseconds in case of ASCII output. Frequencies above a line central frequency, i.e. blue shifted parts are counted as negative velocities and red shifted parts positive.

3.4 The input parameters

3.4.1 Physical cloud parameters

`Background radiation temperature [K]:`
For normal galactic sources, the cosmic background with a temperature of 2.73 K is dominating the external radiation in the wavelength range of the rotational transitions. Looking at early universe objects one should adapt higher values.

`Number of shells for the cloud representation:`
The cloud may be subdivided into a number of shells with different laws for the physical parameters. These shells are only used to to define the input data and have no relation to the radial shells later used in the radiative transfer computation. It is possible to specify discontinuities at shell edges in case of power law regions and file input, however, they should be reserved to the experienced user. The maximum number of shells is restricted to about 180.

`Use shells as power law regions or as data table ?  [0/1]:`
Within the shells one can either specify the radial dependence of the physical parameters by their power law exponents or in terms of a data table where the corresponding shell behaviour is interpolated.

`Radius of the inner core [pc]:`
To avoid the singularity of power laws at zero, the user has to define an inner core. Within this core all parameters are assumed to be constant. Furthermore, the core may consist of an H II region. In case of a data table input the core radius is identical to the innermost table radius.

`Electron density [e/cm^3] and temperature:`
In case of an H II region for the inner core, these parameters have to be specified to compute the continuum absorptivity and emissivity. An electron density 0 means that no central HII region will be assumed.

For each shell the following parameters have to be specified.

`Outer radius of the shell [pc]:`

`Hydrogen density [H2/cm^3]:`

`Temperature [K]:`
The kinetic temperature determines the collision rates and provides a contribution to the local line width.

`Relative molecular abundance [X/H2]:`
Here, possible molecule depletions should be taken into account.

`FWHM of turbulent velocity [km/s]:`
The thermal velocities will be added to get the total kinetic line width.

`Turbulence correlation length [pc]:`
The scale on which the autocorrelation function of turbulence goes to 1/e.

`Systematic radial velocity [km/s]`
: Infall is counted with negative numbers.

In case of power law input, all parameters, except the radius, are characterized by the value at the inner boundary of the shell and a radial exponent. In case of data table input only the values at the corresponding radial points have to be given.

3.4.2 Observational parameters

`Upper level of the transition to be observed:`
: Only one line is considered at a time. Here, we need the number of the upper level in the molecular data file.

`Lower level of the transition to be observed:`
: Number of the lower level in the molecular data file..

`Frequency resolution [km/s]:`
: This might be restricted by the detector or the purpose of the observation.

`Beam width [FWHP in ‘‘]:`
: The beam is considered to be Gaussian.

`Step size for mapping [‘‘]:`
: One typically choses steps in the order of half a beam width.

`Central offset for the first point [‘‘]:`
: In case of observational offsets or composed maps, one may start the radial map not at the centre of the cloud but with a finite offset.

`Number of map points [0=map whole cloud]:`
: A number 1 means just one observation, not a map. When giving a zero, the program automatically computes the number of map points up to the outer radius of the cloud.

`Distance of the cloud [pc]:`
: Needed to translate the physical cloud size into angular units.

3.4.3 Numerical parameters

The same set of numerical parameters can be used to determine the accuracy in the computation of the level populations and in the computation of the emergent line profiles. In a few cases a parameter is interpreted slightly different in the two parts.

`Cut-off for Gaussians in the line emission [sigma]:`
: This one and the next parameter determine at which argument a Gaussian is considered to be zero. For higher arguments the analytic treatment is truncated. The cut-off in line emission determines at which frequencies in radiative transfer the absorptivity/ emissivity is treated as zero, i.e. the background field remains unchanged. The default value implemented in the program is 3.2.

`Cut-off for Gaussians in freq. integration [sigma]:`
: This parameter says where to set the limits in integration over the intensity multiplied with a Gaussian profile. It is applied in the frequency integration to determine the local energy density at each point and in the spatial integration over the telescope beam. The default value is 4.2.

`Resolution in scanning the Gaussians [sigma]:`
: The parameter defines the resolution of the grid for scanning the frequency scale in the integration of the radiative energy density. This frequency grid is locally adapted to be always at least as fine as this resolution factor times the local σ. The default value is 0.55.

`Maximum change factor for the level densities between grid points:`
: The gas densities and level populations are linearily interpolated between the radial grid points. This parameter determines the maximum change of the level densities n_j between the grid points.

In case of steeper changes, additional grid points are added. The factor has to be greater than 1. The default value is 1.3.

In many cases, the reduction of this parameter provides the strongest improvement of the computational accuracy at the costs of a strong increase in the time consumption. It may, therefore, be considered as the most important numerical parameter.

`Hysteresis in the radial grid adjustment [0..1]:`
Since the radial grid is dynamically adjusted to the gradient of the level densities n_j, grid points may be added or removed during the lambda iteration. The hysteresis gives the relative difference between the thresholds for addition and removal of points. It supresses grid oscillations during the iteration. The default value is 0.2.

`Maximum change of the radial velocity between grid points [sigma]:`
Beside the density structure, also the velocity structure may require additional radial grid points. The systematic velocity is linearily interpolated between the grid points. The parameter determines the maximum change of the systematic velocity between two radial grid points. In case of stronger changes, additional grid points are included. The default value is 2.5.

`Maximum relative distance of points on the spatial integration scale:`
This parameter determines the number of displacement values treated in the radiative transfer code. Since the computation of the energy density at a given radial point requires the integration over the z axis (Eq. 10), many points should be given along the z axis. The parameter constrains the maximum relative distance between two points on the z axis within this integration procedure. The default value is 0.25.

`Maximum velocity shift between radiative transfer points [sigma]:`
The radiative transfer code interpolates the emissivity and absorptivity for each frequency linearily within each step. In case of large velocity gradients, this will be inappropriate since the line profiles are widely shifted between two grid points. Here, additional steps have to be included. The parameter gives the maximum length of such a step on the scale of velocity in z direction v_z. It is certainly a good idea to take the same value as for the frequency resolution so that the frequency shift between consequent radiative transfer points is always according to at most one grid spacing at the frequency scale. The default value is 0.55.

`Initial populations [1=thermal,2=no radiation,3=Sobolev, 4:file input]:`
Here, one can determine the first guess for the level populations the lambda iteration starts with. The thermal distribution is an appropriate approach for clouds which are optically thick within at least one line of the molecule considered. The radiation free distribution should be chosen for optically thin clouds. The Sobolev approximation is good for clouds with large velocity gradients. However, I found that, due to the high efficiency of the accelerated lambda iteration method, the influence of the initial guess on the total number of iterations is relatively small. The level populations output file from one model run can be used as an input file for another model. This is especially useful when iteratively fitting observations by scanning the parameter space. The default value is 2.

`Neglection threshold for the level populations:`
The parameter sets the truncation of the equation system for the level populations. Whenever the excitation of one and all higher levels troughout the whole cloud falls below this threshold, the level is neglected in the further computation. According to the machine precision used for the representation of the numbers values below 10^{-14} make no sense. The default value is 10^{-8}.

`Relative accuracy as convergence criterion:`
This is the convergence criterion. It is tested only after a full lambda acceleration cycle. When even the lambda acceleration does not find a larger step size than this accuracy value, convergence is reached. To obtain accuracies better than $2\,10^{-7}$, the program has to be compiled with real*8 numbers instead of the normally used real*4. The default value is 10^{-6}.

Due to the strong nonlinearity of the whole line radiative transfer problem it is not possible to give a general error estimate for a given set of numerical parameters. Tests using a broad range of cloud

parameters have shown, however, that the error in line intensities using the default numerical parameters falls always below 5–6%.

When the command line option `-numdefaults` is given at program invocation, the default numerical parameters are used in all computations. This can avoid the annoying confirmation of the usage of these default values before every computation in cases when one does not plan to change them. Then, it is, however, impossible to use changed numerical parameters within the same program run when it turns out that a given problem is not soluble with these default numerical parameters.

3.5 Error messages and warnings

Due to the static declaration of fields in FORTRAN 77 the possible numbers of grid points on each scale is restricted to the maximum size of the corresponding fields. There are two different precompiled binary versions available. In the normal version the field size for the radial points is set to 180, for the impact parameters to 270, for the frequency points to 223, for the molecular levels to 20 and for the number of radiative transitions to 19. This version should be suitable for most linear molecules. Most errors occur when the combination of cloud properties and numerical parameters requires a grid which exceeds these numbers in one dimension.

If the radiative transfer problem requires some field size larger than the compilation values you will face an error message and either a return to the main input loop (default) or a stop of the program with the exit code 1 (if the option –stoponerror is set in the program call). You may find the following messages:

`Number of radial points insufficient to treat the problem with the required accuracy !`

The gradient in the level densities requires more than 140 radial points either due to a very steep density gradient or due to strong changes in the level populations, e.g. at the edge of very thick clouds. The user will be brought back to the input routines to change some parameters. One might either somewhat increase the numerical parameter characterizing the maximum change of level densities between grid points or in case of extremly differing densities within the cloud substitute regions with steep density gradients by a discontinuity in the gas density.

`Number of ray points insufficient to treat the problem with the required accuracy !`

This error may scarely happen when the user has chosen a very small distance of the grid points on the z integration scale in connection with many radial grid points. It may also appear in the computation of the emergent line profiles when the systematic velocities are very large compared with the local random velocities or a large cloud is mapped with a very fine beam. He will be brought back to the input routines. One should slightly increase the numerical parameter characterizing the relative distance of grid points on the spatial integration scale or, in case of the large velocities, increase the numerical parameter giving the resolution in scanning the Gaussians.

`Number of frequency points insufficient to treat the problem with the required accuracy !`

Looking at the maximum velocities in the cloud and the frequency resolution required either by the local line width in some points of the cloud (first part of the program) or by the observation (second part), more than 223 frequency points would be required. The user will be brought back to the input routines to change some parameters. When the error appears in the first part, i.e. the computation of the level densities, one should increase the numerical parameter characterizing the resolution in scanning the profiles or slightly reduce the numerical parameter for the cut-off in the frequency integration. However, there is no solution when the systematic velocities exceed the local line width by many order of magnitude. If the error appears in the second part, i.e. the computation of the emergent line profiles, one should consider to reduce either the frequency resolution chosen as observational parameter or to reduce the numerical parameter for the cut-off in the frequency integration.

`Numerical singularity occured for the given parameters !`

This message occurs when the general "worst case check" fails indicating that the code is no longer able to compute non-singular values for the level populations. This error typically results from unphysical input parameters like negative cloud temperatures or temperatures high above

the range of valid collision parameters or negative radii. One should carefully inspect the input parameters to detect the true reason.

Other error messages may appear in case of problems with the file input/output.

`I cannot open the input file`
: This message is thrown if the specified input file does not exist or is not readable. With the option –stoponerror, the program stops with exit code 2.

`Format error in input file`
: This message is thrown if the line structure of the input file cannot be recognized, i.e. the order of parameters is wrong or comment lines are missing. With the option –stoponerror, the program stops with exit code 3.

`I cannot open the output file`
: This message is thrown if the output file exists and the –overwrite option is not set, the output file exists and the permissions do not allow to overwrite it or the output file cannot be written because a non-existing directory is specified. With the option –stoponerror, the program stops with exit code 4.

In the input routine for the molecular data file it is checked whether the parameters in the file fit into the internal input fields regarding the number of levels, radiative transitions, temperatures, and collisional rate coefficients. However, if the file contains more levels or transitions than the radiative transfer code can treat, the molecular system will be simply truncated at high energies by virtually removing high levels until all transitions and levels fit into the fields.

There is no explicite test whether the maximum number of atomic levels treated by the program is sufficient for a given problem. I case of doubt one should inspect the output file for the level densities looking at the excitations of the last level. The number of levels can be a serious restriction when treating relatively warm and optically very thick clouds.

A small number of consistency checks is done for the parameter input and the following messages may appear:

`Line overlap detected:  transition - from - to - at - Hz`
: At the moment, the program only checks for overlapping lines but does not yet treat them in radiative transfer. It is just a warning that the results will not be reliable in this case.

`There is no excited transition from level - to level - !`
: This error occurs when the observational parameters select a transition that does either not exist as radiative transition in the molecular data file or that includes a level the excitation of which falls below the neglection threshold of the previous computation of the level densities. A new transition has to be selected.

`Only finite beam widths are possible !`
: The FWHM of the beam has to be > 0.

Nevertheless, most parameters are not checked and strange results or abnormal program termination may occur in case of unphysical or numerically absurd parameters. E.g. one has to take care in the input parameters that the kinetic temperature in all parts of the cloud should be above the cosmic background temperature. Otherwise, the code will probably abort computation at some point.

4 Hints for the application of the code in line fitting

My own tests and the experiences of some colleagues lead to a few recommentations the user of the program should be aware of:

- To get a first feeling one should always start using the default numerical parameters directly implemented in the program. Only in a late stage, the accuracy of the result should be checked by setting stronger limits to the numerical accuracy.

- When fitting a number of lines, the user should always start with a simple one-shell-model (except there is a detailed physical model in background suggesting a more complicated structure). Modifying the parameters within this one shell will already provide the possibility to produce a very wide variety of line shapes and intensity ratios. In complicated models, one can easily get lost within the parameter space. *Remember, in 1-D, there is never only one unique solution to a line fitting problem !*

- To get a reasonable initial guess for the cloud parameters, traditional ways for the determination of the gas temperature and the column density, like the comparison of the line strengths from different isotopes of a considered molecule, should always serve as starting point.

- Whenever the cloud model covers a range of gas densities $> 10^3$ or the density structure is relatively complex, it may happen that one runs out of the number of points available for the radial grid. For smooth density structures, the user should relax the numerical parameter giving the maximum difference in the level densities between two radial points. In case of strongly variing density profiles it may be more appropriate to substitute steep gradients by artificial jumps in the density (The input of discontinuities in the parameters is only possible using file input).

- At the end of a density fit, the user should check the consistency of the temperature estimate with the computed line temperature of the thick line used for the temperature derivation. I found that even quite thick lines ($\tau > 10$) are often not completely thermalized. At least when ^{12}CO is used, this way can provide a much more accurate temperature determination.

- As a first orientation to constrain the turbulence parameter space, one might follow Miesch & Bally (1994) which got correlation lengths of about 0.1 pc for the best resolved clouds. Their values for the Kolmogorov exponent γ giving the radial dependence of the turbulent velocity dispersion range from 0.74 for L1228 down to values around 0.5 for relatively many objects. For a homogeneous isotropic infinite turbulence, there is also a theoretical relation between γ and the correlation length: $\gamma \propto 1/\ln(r_c)$ showing a general anticorrelation.

- The approximation of effective absorption coefficients for a clumpy medium is valid both for clumping in space and velocity space. The used description by the correlation length is according to a homogeneous medium with pure turbulence in velocity space. There is no additional parameter for the filling factor in space in case of a clumpy medium, since this parameter is not independent. Instead, one has to use the clump hydrogen density for the hydrogen density parameter,the molecular abundance has to be reduced by the filling factor and the correlation length has to be increased by one over the filling factor.

5 System requirements

The program is written very close to standard FORTRAN 77 so that most f77 compilers should be able to compile it. It has been tested as it is on DEC Alpha running OSF/1, DEC VAX running Ultrix, SUN Sparc running SunOS, IBM RISC running AIX, HP with HP-UX, and Intel machines running Linux (with f77(f2c), g77, and Absoft-f77). On DEC VAX machines running VMS, four lines in `ltrio.f` have to be changed due to a different record treatment of this operating system compared to the Unixes above. On none of these platforms the compiler or linker should give any warnings. I did not yet succeed to run the code under MS-DOS due to a strange memory management of the Microsoft-Fortran-Compiler/Linker.

The memory usage is mainly determined by the one large field of energy densities at all radial points. In the normal precompiled version it consists of $180 \times 540 \times 19$ real*4 numbers which makes in total about 7.4MB alone. In the extended version it uses $220 \times 660 \times 90$ real*4 numbers, i.e. 52 MB. Adding another 10% for all other fields and the dynamically linked libraries gives a reasonable guess for the total memory consumption.

6 Limitations and bugs

One should keep in mind that a main limitation in all radiative transfer computations comes from the accuracy of available collision rates and the temperature range where they are given. At temperatures

outside of the interval given in the molecular data file the collision coefficients will be linearly extrapolated. Moreover, one should also consider that the maximum number of levels is restricted in the code so that high temperatures combined with high densities might also provide a non-vanishing excitation at levels which are neglected in the computation. The user has to check for his own, that the maximum level number is sufficient.

The program is not designed to fit the situation of a strongly variing turbulence. The gradient in both the absolute value of the turbulent velocity dispersion and in the turbulence correlation length should not exceed the steepest gradient of either the systematic velocity or the density.

Due to the limited number of frequency points the systematic velocities may fall at most one order of magnitude above the random velocities.

At the moment, I know of no bugs in the code.

7 Future plans

New versions of the program are planned to contain:

- a graphical IDL frontend for input/output
- improvement of the treatment of scale-invariant turbulence
- correct treatment of overlapping lines
- interaction with dust continuum radiation
- ...

8 Internals - Structure of the code

Here, I can only give a short overview over the general structure of the code. It shall help as a guide through the source code in case of a deeper inspection of the numerical internals or in case of error debugging. The overview explains the function of the different routines and how they are concatenated within the program. It cannot substitute for a study of the sources which I have tried to comment relatively detailed.

Please, do not perform any essential changes to the sources without contacting me and send me all your suggestions to improve the code or to make it more user-friendly.

8.1 Main program

LINETRANS is the main program. It provides the loops for calling MICRO and LINE, the routines for the computation of the molecular level populations and the observed line profiles.

Both MICRO and LINE start with reading the parameters specified by the user. MICRO then initializes the grids and starts the loop made of the radiative transfer (STPMIC), the lambda iteration (LAMBDA, LEVMIC) and the grid adjustment (ADAPTGRID). In case of convergence the results may be stored (WRITEDENS).

LINE continues with setting up the grids for beam offset and frequency, calls the radiative transfer in one line (STPLINE), the beam integration (INTBEAM) and finally writes the results (WRITELINE).

PRESENT shows the opening and closing screen of the program.

8.2 Input - Output

The routines for the parameter input READPHYS, READNUM, READOBS have the same general structure. One part runs the file input and another one the interactive input. In case of interactive input an inverse version of the file read can be used to produce a parameter file. For the input of numerical parameters (READNUM), there is an additional part so that it is possible to use the built-in default values or to use the parameters from a previous computation if called more than once.

The routine for the input of the molecular data file READSPECIES allows no direct terminal input and contains a translation from input fields which should be large enough to read all reasonable molecular data files and the internal fields which are used in radiative transfer.

WRITEDENS is the routine to store the level populations in a file with the structure described in Sect. 2.2. READDENS reads the populations from such a file. Finally, WRITELINE is used to store the integrated beam temperatures in a file with variable structure. In case of FITS format for line output it calls SAVEFITS which needs to be linked with the FITSIO library and the auxiliary routines PRINTERROR and DELETEFILE from fitsadd.f. The default parameters for the structure of the files written are set in PREDEFINE which reads the command line. The function IOERROR is called in case of file-I/O exceptions. It produces the error messages and determines the change of the program flow.

8.3 Radiative transfer for all lines

These routines perform the radiative transfer computation necessary to obtain the radiation energy density at each point and within each molecular transition.

STPMIC contains the loops for all displacement parameters and all radial points. For each "ray" LEFTKPL is called to initialize the fields for the intensity (with the background radiation - UBACK) and the absorption coefficient at the edge of the cloud.

To carry out the transfer between two radial grid points, at first FIRSTSTEP is called. This subroutine determines the number of real radiation transfer steps necessary between these points to have sufficiently small shifts in the line profiles. If more than one step is required NEXTSTEP is called in a following loop to compute the size of further steps.

TRANSFER is the real radiative transfer routine. It constrains the frequency range where the line profile is non-vanishing and computes the new intensity in this range. In case of small optical depths a linear approximation of the exponential functions (Eq. 3) is done, otherwise the full function will be treated assuming that κ and ϵ show a linear dependence within the integration range. CTRANSFER computes the continuum line transfer within a central H II region. It assumes, that the opacity and source function remain constant within this core.

At each radial point STPMIC calls the integration routines UNYINT and UZINT to compute the energy density.

8.4 Radiative transfer for a single line

The subroutines STPLINE, FIRSTLSTP, NEXTLSTP, LEFTKPL, TRANSFL, and CTRANSFL are reduced versions of STPMIC, FIRSTSTEP, NEXTSTEP, LEFTKP, TRANSFER, and CTRANSFER in the sense that they compute the radiative transfer only in a single line instead of all transitions important for a molecule. They are used to compute the emergent line profiles which are to be observed. No intensity integration is called. The additional routine TAUCENTRAL is used here to give a control output of the central opacity in the line centre.

8.5 Initialization and adjustment of the grids

INITRAD builds the initial radial grid. It checks the gradients in density and velocity and takes as many points as necessary to fulfil the conditions for the maximum changes specified by the numerical parameters. Additional points are included in case of discontinuities detected in the density structure. To assign the physical parameters density, temperature, systematic velocity ... to a radial point VALUES is called which computes these values from the physical input parameters.

INITN initializes the level populations at the different radial grid points according to the chosen first guess approximation. In case of a central H II region, it calls UFROMHII which computes the integrated intensity at a given point due to the continuum radiation from the H II region.

FILEDENS takes both the initial radial grid and the level populations from an procumputed file using READDENS. It thus substitutes both INITRAD and INITN in case of file input for the initial guess.

INITFREQ scans through all points and determines the local line width. From this quantity the maximum frequency/velocity step size for each point is computed. OVERLAP checks for line overlap in the given velocity regime.

ADAPTGRID is responsible for changing the radial grid so that the differences between the points fulfil the conditions for the maximum change of the level population between two points and the maximum velocity difference. It calls ADDPOINTS which checks for the need of additional points and

includes them and REMOVEPOINTS which seeks for superfluid points which will be removed. In case of adding new grid points VALUES is called which computes the gas density and velocities at the new radii.

REMOVELEVELS checks whether some levels may be removed due to negligible population and calls LINFROMLEV in that case to reaarrange the field of transitional coefficients.

The routines MAKEPS and PSFORLINE build the grid of displacement parameters providing the "rays" on which the radiative transfer is calculated. MAKEPS is responsible for the grid used in STPMIC, PSFORLINE builds the grid for STPLINE. Both grids differ. MAKEPS controls that for a given radial point the relative distance of the displacment points on the z scale is below the numerical parameter chosen in NUMREAD (see Sect. 2.3.3). The grid is made as fine as necessary to fulfil this condition. PSFORLINE starts with the radial grid and makes it denser for the displacement grid to have the same accuracy in covering the beam. In contrast to MAKEPS all radial points are used.

8.6 Solution of the balance equations

MATRIXUP is the routine to compute the coefficients and set up the transition matrix. Together with the normalization equation the transition matrix forms the linear system of balance equations (Eq. 8). LUSOLVER solves this system calling LUDCMP for a LU decomposition, LUBKSB for the back substitution and MPROVE for the iterative improvement of the solution.

LEVMIC calls either ULEVMIC or DLEVMIC which store the the energy densities or the level populations at all radial points for three cycles to be used in the lambda acceleration. After three ordinary lambda iteration cycles LAMBDA or DLAMBDA is called which compute a second order lambda acceleration step to the energy densities or the level populations. In case of a sufficiently small acceleration step (convergence criterion) LEVMIC terminates the iteration. HIISMTH corrects the obtained level populations within the HII region to enable a uniform treatment for all grid points.

KAPSRC is a small routine to compute the new absorption coefficients and the source functions from the level populations using the Einstein coefficients from AIJ,BIJ, and BJI. EFFKAP is the routine which obtains the effective absorption coefficients for the turbulent medium from the correlation length and the clump absorptivities. ATAU is the corresponding reduction function. In case of a central HII region, HIIKAP corrects the opacities for the most inner point to get the continuum values.

8.7 The Sobolev approximation

SOBOLEV contains the loop for all radial points and all iterations to determine the local energy densities and level populations in the Sobolev approximation. It computes the velocity gradients which are used by ESCAPE to determine the escape probability in a given direction and its first derivative with respect to n_j. GAUSSOB integrates the escape probabilities over all directions using a Gaussian quadrature, the grid points of which are initialized at the beginning by GAULEG. From the integrated escape probabilities SOBOLEV computes the local energy densities which are used by SOBMATRIX to obtain the level populations.

SOBMATRIX is the routine to set up the differential form of the balance equation system used for the Newton-Raphson approach to the local level populations (Eq. 23). It calls AMATRIX to get the original form of the matrix (used on the right hand side of the system) and DMATRIX for the differential form (for the left hand side). The system is resolved by LUSOLVER. SOBKAPSRC is the routine to get the absorptivities and the source function and their derivatives from the level populations. SOBEFFKAP computes the effective absorption coefficient and its derivatives for a clumpy turbulent medium.

8.8 Intensity integration

UNYINT sets the limits for the frequency integration at each radial point and multiplies the intensities with the local line profile (Eq. 10). Then, it calls QUINTGAUSS, a simple equidistant integration routine, which carries out the frequency integration. The line profile is computed by PHI.

UZINT prepares the fields for the z integration (Eq. 10). In ranges with very small steps it calls the linear integration routine DINTLIN, for larger steps it calls the cubic spline integration routine DINTCUB. DINTCUB itself makes use of the cubic spline interpolation routine SPLINE.

INTBEAM is responsible for the intensity integration over the telescope beam. It computes the range of the cloud covered by a Gaussian beam and calls SPECERR for the ϕ integration and finally DINTCUB, the cubic integration program, for the p integration. SPECERR computes an numerical approximation to the ϕ integral within Eq. (24) which is independent from the intensity.

TRANSLAT is a small routine translating the angular beam width of the telescope into the projected beam width on the cloud.

8.9 Molecular constants

The following routines contain the full "physics" of the problem. All physical quantities are specified within them. All previous routines only contain "numerics". The fields used here are read in advance by READSPECIES.

INITEINSTEIN looks which molecule shall be considered and initializes the fields containing the molecule specific quantities, especially $a_{j,j-1}$, $b_{j,j-1}$, and ν_0. The functions AIJ, BIJ, BJI and CFREQ simply read the tables of $A_{j,l}$, $B_{j,l}$, and ν_0 and provide the results as function values.

PROBTHERM, VTHERM, UBACK also base on the tables produced by INITEINSTEIN. These functions provide the population of a distinct level in case of thermal distribution, compute the mean thermal velocity of the molecules at a given temperature, and determine the intensity of the cosmic background radiation at the frequencies of the molecule transitions. HIIINIT computes the continuum opacity and source function for all transitions of the considered molecule from the table of frequencies and the electron density and temperature within a given HII region.

CIJ obtains the collision rate coefficients at a given temperature for the transition of interest. It calls ILOCAT to find the appropriate table entries and interpolates the values for the given temperature.

9 Source files

The following files contain the sources for the different routines:

`ltr.f`: LINETRANS, MICRO, LINE, PHI, TRANSLAT, PRESENT

`ltrio.f`: READPHYS, READNUM, READOBS, WRITEDENS, READDENS, WRITELINE, SAVEFITS,IOERROR, PREDEFINE

`initial.f`: INITRAD, INITN, FILEDENS, INITFREQ, ADAPTGRID, ADDPOINTS, REMOVEPOINTS, VALUES, UFROMHII, OVERLAP

`stpdiff.f`: STPMIC, FIRSTSTEP, NEXTSTEP, LEFTKP, TRANSFER, MAKEPS

`stpline.f`: STPLINE, FIRSTLSTP, NEXTLSTP, LEFTKPL, TRANSFL, PSFORLINE, TAUCENTRAL

`matrix.f`: LEVMIC, ULEVMIC, DLEVMIC, LAMBDA, DLAMBDA, REMOVELEVELS, LINFROMLEV, KAPSRC, EFFKAP, ATAU, HIIKAP, HIISMTH, MATRIXUP, LUSOLVER, LUDCMP, LUBKSB, MPROVE

`sobolev.f`: SOBOLEV, ESCAPE, GAUSSOB, GAULEG, SOBMATRIX, SOBKAPSRC, SOBEFFKAP

`einstein.f`: AIJ, BIJ, BJI, CFREQ, PROBTHERM, VTHERM, PLANCK, UBACK, HIIINIT

`collrate.f`: READSPECIES, CIJ, ILOCAT

`ltrinteg.f`: INTBEAM, SPECERR, UNYINT, QUINTGAUSS, UZINT, DINTCUB, DINTLIN, SPLINE

`fitsadd.f`: PRINTERROR, DELETEFILE

10 Auxiliary programs

I have written some small auxiliary programs, mainly for testing purposes around the SimLine code. The program TESTCLOUD reads the level populations and energy densities from a data file produced in the first part of SimLine and computes the excitation temperatures, radiation temperatures and few additional quantities. The input file must have the structure as produced by WRITEDENS, the output file is self explanatory.

The program CLINE reduces the two dimensional data set as produced by WRITELINE and extracts the profile for zero beam offset, i.e. the resulting file contains only one line profile. This may be suitable for users which do not have a sophisticated graphics front end as GREG or IDL.

11 Acknowledgements

A direct precursor of this program was written by Endrik Krügel (MPIfR Bonn). Basing on his code of about 1000 lines, I could set up this more comprehensive and flexible program. The subroutines CIJCOH2 and ILOCAT are still very similar to the original code.

The local statistical treatment of clumping or turbulence roughly follows the formalism derived by H.M. Martin, D.B. Sanders & R.E. Hills (1985). The accelerated lambda method was published by Lawrence Auer in 1987. The LU decomposition method, the Gaussian quadrature, and the spline interpolation algorithm were taken from "Numerical Recipies" (Press, Flannery, Teukolsky & Vetterling 1986).

12 References

Auer L.H. 1987, In: Kalkofen W. (Ed.), Numerical Radiative Transfer, Cambridge, p. 101

Flower D.R., Launay J.M. 1985, MNRAS 214, 271

Fuller G.A., Myers P.C. 1992, ApJ 384, 523

Green S. 1975, ApJ 201, 366

Green S., Chapman S. 1978, ApJSS 37, 169

Heaton B.D., Little L.T., Yamashita T., Davies S.R., Cunningham C.T., Monteiro T.S. 1993, A&A 278, 238

Krügel E., Chini R. 1994, A&A 287, 947

Le Bourlot J. 1991, A&A 242, 235

Martin H.M., Sanders D.B., Hills R.E. 1984, MNRAS 208, 35

Miesch M.S., Bally J. 1994, ApJ 429, 645

Monteiro T.S. 1985, MNRAS 214, 419

Press W.H., Flannery B.P., Teukolsky S.A., Vetterling W.T. 1986 Numerical Recipies (Fortran Version), Cambridge

Turner B.E., Chan K., Green S., Lubowich D.A., ApJ 399, 114