Bluff your way into Cloudy modeling

Henrik Spoon

May 8, 2003

Some fool proof recipes for using MICE

— the MPE IDL CLOUDY Environment —

Cloudy version

This version of the MICE manual is written to comply with CLOUDY version 96.00 and its beta versions.

MICE colors

To load the MICE color table type `micecol`. If you prefer a grey scale color table, type `micegrey` instead.

MICE plotting window

The MICE plotting window is created by typing `micewin`.

How to compute a single CLOUDY model

Requested: a CLOUDY model for a supergiant of T_{eff}=45000 K and Z=0.02 from the Pauldrach-2000 library, and nebular parameters *HDEN 3.0* and *IONIZATION -2.5*. Results should be obtained for all computed zones. Continua and emissivities should be obtained as well.

NB: □ denotes a mandatory blank!

(Type 1)

1. **Create MICE input file** *mice.input***:**
   External□Paul00□(s45,0.02)
   ****
   ****
   HDEN 3.0
   IONIZATION -2.5
   STOP TEMPERATURE=100K
2. **Compute a single CLOUDY model:**
   `IDL>Make_CloudIn,'mice.input',nzones=1`
   `LINUX>./cloudy.sequence`
   `IDL>@mice.sequence`

3. **The resulting CLOUDY model is stored in a CLI structure named `Res`.**
4. **In addition, the structure is saved as an IDL save file named** *Res.Save*.

How to compute CLOUDY models for a grid of SEDs

Requested: CLOUDY models for seven central source SED shapes. CLOUDY parameters are not varied. The seven SED shapes are the following:

– Pauldrach-1997 model for T_{eff}=45000 K, log g=4.0, Z=0.02
– Pauldrach-2000 model for T_{eff}=45000 K, log g=3.8, Z=0.02
– Schaerer-1996 model 3 from track C at Z=0.004
– Schaerer-1996 model 4 from track D at Z=0.020
– Kurucz-1994 model for T_{eff}=35000 K, log g=4.5 at ZSolar=1 (Z=0.020·10^{ZSolar}=0.20)
– Hybrid model for T_{eff}=62500 K, log g=4.2 at Z=0.02
– STARS97 model *t65_u050_s052.dat*
– Blackbody SED *B75.dat* for T_{eff}=75000 K. The SED file is created in the following way:
– `IDL>B75=BlackBody(75000,OutName='B75.dat')`

1. **Create MICE input file** *mice.input*: **(Type 2)**
   External␣Paul97␣(45,4.0,0.02)
   External␣Paul00␣(d45,0.02)
   External␣Sch96␣(c3,0.004)(D4,0.02)
   External␣Kur94␣(35,4.5,1)
   External␣Hybrid␣(62.5,4.2,0.02)
   External␣Stars97␣(t65_u050_s052.dat)
   External␣Model␣(B75.dat)
   ****
   ****
   HDEN 3.0
   IONIZATION -2.5
   STOP TEMPERATURE=100K
2. **Compute the set of CLOUDY models:**
   `IDL>Make_CloudIn,'mice.input'`
   `LINUX>./cloudy.sequence`
   `IDL>@mice.sequence`
3. **The resulting set of seven CLOUDY models is stored in a one-dimensional CLI structure named `Res`.**
4. **In addition, the structure is saved as an IDL save file named** *Res.Save*.

How to compute CLOUDY models for a grid of CLOUDY parameters

Requested: CLOUDY models for a dwarf star of T_{eff}=45000 K, and Z=0.01 from the Pauldrach-2000 library. A 2-dimensional grid of nebular conditions is to be explored:

– HDEN={2.0,2.5,3.0,3.5,4.0,4.5,5.0}
– IONIZATION={-3.0,-2.5,-2.0,-1.5,-1.0,-0.5,0.0,+0.5}

1. **Create MICE input file** *mice.input*: **(Type 3)**
   External␣Paul00␣(d45,0.01)
   ****

HDEN␣(2.0)(2.5)(3.0)(3.5)(4.0)(4.5)(5.0)
IONIZATION␣(-3.0)(-2.5)(-2.0)(-1.5)(-1.0)(-0.5)(0.0)(+0.5)
****
STOP TEMPERATURE=100K

2. **Compute the set of CLOUDY models:**
   `IDL>Make_CloudIn,'mice.input'`
   `LINUX>./cloudy.sequence`
   `IDL>@mice.sequence`
3. **The resulting set of 7×8 CLOUDY models is stored in a 2-dimensional CLI structure named** `Res`.
4. **In addition, the structure is saved as an IDL save file named** *Res.Save.*

How to compute CLOUDY models for a grid of SEDS and CLOUDY parameters

Requested: CLOUDY models for several Kurucz central source SED shapes. A bolometric luminosity of 10^{39} erg/s is assumed. Three CLOUDY parameters are to be varied:

– HDEN={3.0,3.5,4.0,4.5,5.0} (hydrogen density)
– RLAW={0,-0.5,-1,-1.5} (density power law index)
– INNER RADIUS={17,17.3,17.6,17.9} (cloud inner radius)

Because of the dimensions of this grid, we chose to not store the CLOUDY continua and the computed emissivities in the CLI structure. The options `/NoCont` and `/NoEmissiv` of routine *Make_Cloudin* control this. Another way to limit the CLI size is to economize on the species in the line list. The default linelist is therefore replaced by a file *Cloudy.ref* containing less line species.

1. **Create MICE input file** *mice.input***:** **(Type 4)**
   External␣Kur94␣(30,4.5,0)(32,4.5,0)(34,4.5,0)(36,4.5,0)(38,4.5,0)
   External␣Kur94␣(40,4.5,0)(42,4.5,0)(44,4.5,0)(46,4.5,0)(48,4.5,0)(50,5.0,0)
   ****
   HDEN␣(3.0)(3.5)(4.0)(4.5)(5.0)
   RLAW␣(0)(-0.5)(-1)(-1.5)
   INNER␣RADIUS␣(17)(17.3)(17.6)(17.9)
   ****
   LUMINOSITY TOTAL 39
   STOP TEMPERATURE=100K
2. **Compute the set of CLOUDY models:**
   `IDL>Make_CloudIn,'mice.input',/NoCont,Ref='Cloudy.Ref'`
   `LINUX>./cloudy.sequence`
   `IDL>@mice.sequence`
3. **The resulting set of 11×5×4×4 CLOUDY models is stored in a 4-dimensional CLI structure named** `Res`.
4. **In addition, the structure is saved as an IDL save file named** *Res.Save.*

Grid parameter template

Below an overview is given of all parameters that can take part in MICE grid computations. Some notes:

– No more than six parameters can be varied at a time.
– Any collection of `External` or `Internal` SEDs together account for just one grid dimension.
– After the `Internal` keyword, any CLOUDY internal SED can be requested.
– `DENSLAW` allows you to state combinations of the CLOUDY commands HDEN and POWER, which together define a density power law.
– `OUTER RADIUS` can only be varied when `INNER RADIUS` is varied as well. The largest value for `INNER RADIUS` must be smaller than the smallest value for `OUTER RADIUS`.
– `BOTH RADII` allows you to state combinations of the inner and outer radius of CLOUDY command RADIUS.
– `USER1` can be used to vary any type of CLOUDY command, as long as that command can be stated on just a single line.
– `USER2`: just like `USER1`.

Template MICE input file *mice.input*:

```
External␣Hybrid␣(21.5,2.4,0.02)(62,4.2,0.02)
External␣Kun94␣(35,3.2,0.006)(40,3.5,0.02)
External␣Kur94␣(30,4.5,0.0)(32,4.5,-0.5)
External␣Paul97␣(30,3.0,0.02)(50,4.0,0.02)
External␣Paul00␣(d55,0.02)(s45,0.01)
External␣Sch96␣(C3,0.004)(D4,0.02)
External␣Sel96␣(35,3.2)(50,4.0)
External␣Stars96␣(t65_u050_s052.dat)(t73_u100_s061.dat)
External␣Stars97␣(t73_u070_s078.dat)(t64_u120_s065.dat)
External␣Model␣(B75.dat)(Merged.dat)
Internal␣TABLE POWER LAW
Internal␣TABLE CRAB
Internal␣TABLE STAR KURUCZ 39000
Internal␣TABLE STAR KURUCZ 44000
*****
HDEN␣(3.0)(3.5)(4.0)
RLAW␣(-0.5)(-1)(-1.5)
DENSLAW␣(3,-0.53)(3.1,-1)(3.674,-1.5)
IONIZATION␣(-1.0)(-0.5)(0.0)(+0.5)
Q(H)␣(49.3)(50.2)
PHI(H)␣(12.876)(11.3)
LUMINOSITY TOTAL␣(38.1)(39.5)(40.2)
INNER␣RADIUS␣(17)(17.3)(17.6)
OUTER␣RADIUS␣(17.7)(17.8)(17.9)
BOTH␣RADII␣(17,17.9)(17.8,18.2)(17.9,18.2)
USER1␣GRAINS␣ORION␣(0.1)(0.5)(1.0)(2.0)(4.0)
USER2␣ELEMENT␣OXYGEN␣SCALE␣(0.001)(0.01)(0.1)(1.0)(10)
*****
STOP TEMPERATURE=100K
```

Line list template

CLOUDY computes line strengths for a vast amount of spectral lines. From this *mer à boire* MICE selects the results for those spectral lines that are listed in the MICE file *Cloudy_c96.linelist*.
Below an example is given of the format this file should have. Column 1 contains the name of the species emitting the line; column 2 the rest wavelength in microns; and finally, in

column 3, the CLOUDY line ID (the line ID as used in the CLOUDY output file *results.out*). Under default conditions, `Make_CloudIn` will use the line list contained in file *Cloudy_c96.linelist*. A personal line list *Personal.linelist* can be read in by typing:
`IDL>Make_CloudIn,'mice.input',Ref='Personal.linelist'`
NOTE: do not use the TABS key while editing this file, nor end the file with an empty line!

```
---1----␣-----2----␣-----3-----

H          0.656300 TOTL  4861A  # 0
[CII]    157.740900 C   2 157.6m  # 1
[OI]      63.183705 O   1 63.17m  # 2
[OI]     145.525439 O   1 145.5m  # 3
[OII]      0.372848 TOTL  3727A  # 4
[OII]      0.732148 TOTL  7325A  # 5
[OIII]     0.436444 TOTL  4363A  # 6
[OIII]     0.496029 O   3  4959A  # 7
[OIII]     0.500824 O   3  5007A  # 8
[OIII]    51.814500 O   3 51.80m  # 9
[OIII]    88.356000 O   3 88.33m  #10
[OIV]     25.890300 O   4 25.88m  #11
[NeII]    12.813550 Ne  2 12.81m  #12
[NeIII]   15.555100 Ne  3 15.55m  #13
[NeIII]   36.013500 Ne  3 36.00m  #14
Etc.
```

Advanced MICE usage

The output produced by CLOUDY and STOREY is stored into the structure that for historical reasons is known as the "CLI" structure. This structure is the starting point for many MICE applications.
Although MICE plotting routines exist for many parameters of interest, there will always be need to access parameters from the CLI directly. It is therefore useful to know what information is contained in the CLI structure, so that you can access it yourself.
A printout of the CLI structure is contained in file *cli.struct*. Short descriptions of the CLI tags are given in the files *ias_define_cli**.pro* (where ** is the version number of the current CLI structure) and *read_cloudy.pro*.
The basic structure of the CLI tree consists of three parts: `grid`, `general` and `data`. `grid` contains information about the kind of grid you have computed, `general` contains general information about each of the models of the grid, and `data` has information for individual shells ("zones") of each grid model.

To view the grid parameters that are varied, type:
`IDL>print,res.grid.dimname`
For a type-3 CLI the grid parameters varied are HDEN and IONIZATION.

The size of the grid is found by typing:
`IDL>print,res.grid.dimsize`
The size of the HDEN-IONIZATION grid is 7×8.

To find the 7 grid values for HDEN, type:
`IDL>print,res.grid.dimvalue(0,0:6,0)`

To find the 8 grid values for IONIZATION, type:
`IDL>print,res.grid.dimvalue(0,0:7,0)`

With this information you can for instance check the value of the ionization parameter as computed by CLOUDY for model (HDEN,IONIZATION)=(2,3). Type:
`IDL>print,res.general(2,3).incident.ioniz_1r`

To print the 12.81μm [NeII] flux (20th entry in the *cloudy_c90.ref* linelist) as a function of the ionization parameter for the fourth HDEN grid value, type:
`IDL>print,res.data(1,3,*).list(19).flux`
The first dimension of `res.data` is not a grid dimension but a CLOUDY zone number. Since for a grid only the results for the innermost (zone=0) and the outermost (zone=1) zones are stored, you need to state the outer zone explicitly.

To get the radial distance [cm] from the inner face of the cloud to the outer edge of each zone for a type-1 CLI, type:
`IDL>print,res.data(*).geom.depth`
For a type-3 CLI (with only the results for the inner- and outermost zones), type:
`IDL>print,res.data(*,2,3).geom.depth`
This gives you the result for grid model (HDEN,IONIZATION)=(2,3).

How to check for CLOUDY warning messages within a CLI struct

Every computed CLOUDY model is acompanied by some modeling "messages", telling the user what assumptions were been made and which stopping criterium caused the code to stop computations. Especially for large grids it is impossible to keep track of all these messages.
The routine *Find_Cli_Warnings* allows the user to search a CLI structure for CLOUDY warning messages, CLOUDY caution messages and MICE warning messages.

1. **Create** ***mice.input*** **file of type 1,2,3 or 4.**
2. **Run** *Make_Cloudin*
3. **Run the** ***mice.sequence*** **IDL script.**
4. **Search CLI struct for warning messages:**

4a. `IDL>Find_Cli_Warnings,Res,Outname='Warning.List'`
Search the CLI struct for CLOUDY and MICE warning messages, display them on screen and save them to file

4b. `IDL>Find_Cli_Warnings,Res,/Cautions`
Search the CLI struct for CLOUDY warnings and cautions, as well as for MICE warnings

Making hardcopies

The MICE package comes with two routines which will help you to produce hardcopy versions of MICE plots. Before giving the plotting command just type `starthcp`. After you have given the plotting command type `stophcp`. You should find a file named *idl.ps* in your working directory.
Note that all plotting routines are equipped with the following two keywords: `bw` and `thick`. By setting keyword `bw` the default colors for plot background (black) and axes (white) are

inverted. This will save a lot of toner! Setting the keyword `thick` to 2 or 3 increases the thickness of all elements of your plot.

Visualize CLOUDY continua

1. **Create *mice.input* file of type 1,2,3 or 4.**
2. **Run *Make_Cloudin* without option** `/NoCont`
3. **Run the *mice.sequence* IDL script.**
4. **Initialize MICE plot window (load color table & plot window definition):**
   `IDL>MiceWin`
   `IDL>MiceCol`
5. **Use *Plot_Cont* to visualize the CLOUDY continua.**

5a. **For "type 1" (0-dim.) CLIs:**
   `IDL>Plot_Cont,Res`

5b. **For "type 2" (1-dim.) CLIs:**
   `IDL>Plot_Cont,Res,Sed=3`
   Display results for fourth SED grid model contained in the CLI

5c. **For "type 3" (2-dim.) CLIs:**
   `IDL>Plot_Cont,Res,Hden=6,Ion=-1.5`
   Display results for seventh HDEN grid value and IONIZATION=-1.5

5c. **For "type 4" (4-dim.) CLIs:**
   `IDL>Plot_Cont,Res,Sed=10,Hden=2,RLaw=-1,Inr=17.6`
   Display results for tenth SED and third HDEN grid value, density POWER law index -1 and inner RADIUS 17.6

Visualize ionization(r)

1. **Create *mice.input* file of type 1 (all zone results are read in).**
2. **Run *Make_Cloudin***
3. **Run the *mice.sequence* IDL script.**
4. **Initialize MICE plot window (load color table & plot window definition):**
   `IDL>MiceWin`
   `IDL>MiceCol`
5. **Use *Plot_Ion* to visualize the ionization structure of Neon.**
   `IDL>Plot_Ion,Res,'Ne'`

Visualize linestrength(r)

1. **Create *mice.input* file of type 1 (all zone results are read in).**
2. **Run *Make_Cloudin***
3. **Run the *mice.sequence* IDL script.**
4. **Initialize MICE plot window (load color table & plot window definition):**
   `IDL>MiceWin`
   `IDL>MiceCol`
5. **Use *Plot_Emissiv* to visualize the incremental line formation of the Neon emission lines:**
   `IDL>Plot_Emissiv,Res,'Ne',/Intens`

Visualize emissivity(r)

1. **Create** *mice.input* **file of type 1 (all zone results are read in).**
2. **Run** *Make_Cloudin* **without option** `/NoEmissiv`
3. **Run the** *mice.sequence* **IDL script.**
4. **Initialize MICE plot window (load color table & plot window definition):**
   `IDL>MiceWin`
   `IDL>MiceCol`
5. **Use** *Plot_Emissiv* **to visualize the emissivity of Neon:**
   `IDL>Plot_Emissiv,Res,'Ne'`

Quick look of $\mathbf{T}_{el}$(r), $\mathbf{n}_{el}$(r), $\mathbf{N}_H$(r), $\mathbf{P}_{gas}$(r), $\mathbf{P}_{rad}$(r), $\mathbf{F}_{H\beta}$(r), τ_{912}^{abs}(r), [$\mathbf{H}^0$/H](r) and [$\mathbf{H}^+$/H](r)

1. **Create** *mice.input* **file of type 1 (all zone results are read in).**
2. **Run** *Make_Cloudin*
3. **Run the** *mice.sequence* **IDL script.**
4. **Initialize MICE plot window (load color table & plot window definition):**
   `IDL>MiceWin`
   `IDL>MiceCol`
5. **Use** *Plot_Vitals* **to visualize $\mathbf{T}_{el}$(r), $\mathbf{n}_{el}$(r), $\mathbf{N}_H$(r) and $\mathbf{F}_{H\beta}$(r):**
   `IDL>Plot_Vitals,Res`
6. **Use** *Plot_Vitals* **to visualize $\mathbf{P}_{rad}$(r), [$\mathbf{H}^0$/H](r), [$\mathbf{H}^+$/H](r) and τ_{912}^{abs}(r):**
   `IDL>Plot_Vitals,Res,Set=[6,9,10,11]`

Visualize line flux or line ratio as a function of one grid parameter

1. **Create** *mice.input* **file of type 2,3 or 4.**
2. **Run** *Make_Cloudin*
3. **Run the** *mice.sequence* **IDL script.**
4. **Initialize MICE plot window (load color table & plot window definition):**
   `IDL>MiceWin`
   `IDL>MiceCol`
5. **Use** *Plot_1D_Ratios* **to visualize the dependency of some SWS line ratios on one grid parameter:**

5a. **For "type-2" (1-dim.) CLIs:**
   `IDL>Plot_1D_Ratios,Res`
   Displays the dependency of line ratios on SED grid model

5b. **For "type-3" (2-dim.) CLIs:**
   `IDL>Plot_1D_Ratios,Res,HDen=3.5`
   Displays the dependency of line ratios on the IONIZATION parameter value, evaluated at HDEN=3.5
   `IDL>Plot_1D_Ratios,Res,Ion=4`
   Displays the dependency of line ratios on the HDEN parameter value, evaluated at the fifth IONIZATION grid value

5c. **For "type-4" (4-dim.) CLIs:**
   `IDL>Plot_1D_Ratios,Res,Sed=8,HDen=3.5,RLaw=-0.5`
   Displays the dependency of line ratios on the inner RADIUS, evaluated for the ninth

SED, at HDEN=3.5 and density POWER law index -0.5
`IDL>Plot_1D_Ratios,Res,HDen=2,RLaw=-1.0,Inr=17.6`
Displays the dependency of line ratios on the SED shape, evaluated at the third HDEN grid value, density POWER law index -1.0 and an inner RADIUS of 17.6

6. **Use *Plot_1D_Fluxes* to visualize the dependency of line fluxes or line ratios on one grid parameter:**
(In the examples below the numbers 10 and 11 refer to the 11th and 12th line species contained in the *Cloudy_C96.linelist* linelist)

6a. **For "type-2" (1-dim.) CLIs:**
`IDL>Plot_1D_Fluxes,Res,11`
Displays the dependency of the linestrength on the SED grid model
`IDL>Plot_1D_Fluxes,Res,11,id2=10`
Displays the dependency of the 11/10 line ratio on the SED grid model

6b. **For "type-3" (2-dim.) CLIs:**
`IDL>Plot_1D_Fluxes,Res,11,HDen=3.5`
Displays the dependency of the linestrength on the IONIZATION parameter value, evaluated at HDEN=3.5
`IDL>Plot_1D_Fluxes,Res,11,id2=10,HDen=3.5`
Displays the dependency of the 11/10 line ratio on the IONIZATION parameter value, evaluated at HDEN=3.5

6c. **For "type-4" (4-dim.) CLIs:**
`IDL>Plot_1D_Fluxes,Res,11,Sed=8,HDen=3.5,RLaw=-0.5`
Displays the dependency of the linestrength on the inner RADIUS, evaluated for the ninth SED, at HDEN=3.5 and density POWER law index -0.5
`IDL>Plot_1D_Fluxes,Res,11,id2=10,Sed=8,HDen=3.5,RLaw=-0.5`
Displays the dependency of the 11/10 line ratio on the inner RADIUS, evaluated for the ninth SED, at HDEN=3.5 and density POWER law index -0.5

Visualize line flux or line ratio as a function of two grid parameters

1. **Create *mice.input* file of type 3 or 4.**
2. **Run *Make_Cloudin***
3. **Run the *mice.sequence* IDL script.**
4. **Initialize MICE plot window (load color table & plot window definition):**
`IDL>MiceWin`
`IDL>MiceCol`
5. **Use *Plot_2D_Fluxes* to visualize the dependency of some SWS line ratios on one grid parameter:**
(In the examples below the numbers 10 and 11 refer to the 11th and 12th line species contained in the *Cloudy_C96.linelist* linelist)

5a. **For "type-3" (2-dim.) CLIs:**
`IDL>Plot_2D_Fluxes,Res,11,id2=10`
Displays the dependency of the 11/10 line ratio on both the IONIZATION parameter value and the HDEN value

5b. **For "type-4" (4-dim.) CLIs:**
`IDL>Plot_2D_Fluxes,Res,11,id2=10,HDen=3.5,RLaw=-0.5`
Displays the dependency of the 11/10 line ratio on the inner RADIUS, evaluated at HDEN=3.5 and density POWER law index -0.5
`IDL>Plot_2D_Fluxes,Res,11,id2=10,RLaw=-1.0,Inr=17.6,/NoSurf`
Displays the dependency of the 11/10 line ratio on the SED shape, evaluated

at density POWER law index -1.0 and an inner RADIUS of 17.6. Data is shown in a contour plot

Visualize the dependency of two line ratios on one grid parameter

As an example we will generate a diagnostic diagram to determine the electron gas density from the line ratios [SIII] 33.5/18.7 μm and [NeIII] 36.0/15.5 μm.

1. **Create** *mice.input* **file of type 2, 3 or 4** in which HDEN is varied between 1.0 and 6.0 in steps of 0.5.
2. **Run** *Make_Cloudin*
3. **Run the** *mice.sequence* **IDL script.**
4. **Initialize MICE plot window (load color table & plot window definition):**
   `IDL>MiceWin`
   `IDL>MiceCol`
5. **Use** *Plot_ratio_vs_ratio* **to visualize the dependency of two line ratios on one grid parameter:**
   (In the examples below the numbers 22, 23, 46, 47 refer to the [NeIII] 15.5 μm, [NeIII] 36.0 μm, [SIII] 33.5 μm and [NeIII] 18.7 μm lines in the *Cloudy_C96.linelist* linelist)

5a. **For "type-2" (1-dim.) CLIs:**
   `IDL>Plot_ratio_vs_ratio,Res,23,22,47,46`
   Show the plot of the line ratio species-23/species-22 on the X-axis versus the line ratio species-47/species-46 on the Y-axis as a function of one grid parameter

5b. **For "type-3" (2-dim.) CLIs:**
   `IDL>Plot_ratio_vs_ratio,Res,23,22,47,46,Sed=6`
   Show the plot of the line ratio species-23/species-22 on the X-axis versus the line ratio species-47/species-46 on the Y-axis as a function of one grid parameter. The results are shown for the seventh SED

5c. **For "type-4" (3-dim.) CLIs:**
   `IDL>Plot_ratio_vs_ratio,Res,23,22,47,46,Sed=6,Ion=0.5`
   Show the plot of the line ratio species-23/species-22 on the X-axis versus the line ratio species-47/species-46 on the Y-axis as a function of one grid parameter. The results are shown for the seventh SED and IONIZATION=0.5

Visualize line spectrum

1. **Create** *mice.input* **file of type 1, 2, 3 or 4.**
2. **Run** *Make_Cloudin*
3. **Run the** *mice.sequence* **IDL script.**
4. **Initialize MICE plot window (load color table & plot window definition):**
   `IDL>MiceWin`
   `IDL>MiceCol`
5. **Use** *Plot_LineSpec* **to visualize the linestrengths for all lines listed in** *Mice_Tables:Cloudy_C96.linelist***:**

5a. **For "type-1" (0-dim.) CLIs:**
   `IDL>Plot_LineSpec,Res`

5b. **For "type-2" (1-dim.) CLIs:**
   `IDL>Plot_LineSpec,Res,Sed=6`
   Show the results for the seventh SED

5c. **For "type-3" (2-dim.) CLIs:**
`IDL>Plot_LineSpec,Res,HDen=3.5,Ion=0.5`
Show the results for HDEN=3.5 and IONIZATION=0.5

5d. **For "type-4" (4-dim.) CLIs:**
`IDL>Plot_LineSpec,Res,Sed=9,HDen=2,RLaw=-1,Inr=17.6`
Show the results for the tenth SED, the third HDEN value, density POWER law index -1 and an inner RADIUS of 17.6

Visualize "Case-B" hydrogen linestrengths

1. **Create** *mice.input* **file of type 1, 2, 3 or 4.**
2. **Run** *Make_Cloudin*
3. **Run the** *mice.sequence* **IDL script.**
4. **Initialize MICE plot window (load color table & plot window definition):**
`IDL>MiceWin`
`IDL>MiceCol`
5. **Use** *Plot_CaseB* **to visualize the "case-B" hydrogen recombination linestrengths**

5a. **For "type-1" (0-dim.) CLIs:**
`IDL>Plot_CaseB,Res`

5b. **For "type-2" (1-dim.) CLIs:**
`IDL>Plot_CaseB,Res,Sed=6`
Show the results for the seventh SED

5c. **For "type-3" (2-dim.) CLIs:**
`IDL>Plot_CaseB,Res,HDen=3.5,Ion=0.5`
Show the results for HDEN=3.5 and IONIZATION=0.5

5d. **For "type-4" (4-dim.) CLIs:**
`IDL>Plot_CaseB,Res,Sed=9,HDen=2,RLaw=-1,Inr=17.6`
Show the results for the tenth SED, the third HDEN value, density POWER law index -1 and an inner RADIUS of 17.6

Visualize the nebular ionization

1. **Create** *mice.input* **file of type 1, 2, 3 or 4.**
2. **Run** *Make_Cloudin*
3. **Run the** *mice.sequence* **IDL script.**
4. **Initialize MICE plot window (load color table & plot window definition):**
`IDL>MiceWin`
`IDL>MiceCol`
5. **Use** *Plot_Ionization* **to visualize the nebular ionization in a specific zone:**

5a. **For "type-1" (0-dim.) CLIs:**
`IDL>Plot_Ionization,Res,Zone=57`
Show the nebular ionization in zone 57

5b. **For "type-2" (1-dim.) CLIs:**
`IDL>Plot_Ionization,Res,Sed=6`
Show the nebular ionization at the outer face of the nebula for the seventh SED

5c. **For "type-3" (2-dim.) CLIs:**
`IDL>Plot_Ionization,Res,Zone=1,HDen=3.5,Ion=0.5`
Show the nebular ionization at the inner face of the nebula for HDEN=3.5 and IONIZATION=0.5

5d. **For "type-4" (4-dim.) CLIs:**
`IDL>Plot_Ionization,Res,Sed=9,HDen=2,RLaw=-1,Inr=17.6`
Show the nebular ionization at the outer face of the nebula for the tenth SED,

the third HDEN value, density POWER law index -1 and an inner RADIUS of 17.6

Compute F_{bol}, L_{bol}, F_{range}, L_{range}, N_{range}^{phot} for any SED

The routine *FluxFrac* computes bolometric as well as partial range energy fluxes, photon fluxes and luminosities for any SED. The SED to be assessed has to be available either as MICE internal SED struct or as MICE SED file.

1. **Examples showing how to create MICE SED structs and MICE SED files:**
1a. `IDL>Sed3540=Read_Kur(35000,4.0,Out='Kur3540.sed')`
Obtain the Kurucz-1994 SED for T_{eff}=35kK, Log g=4.0 and Z=0.02
1b. `IDL>Sed4045=Read_Kur(40000,4.5,ZSolar=-0.5,Out='Kur4045.sed')`
Obtain the Kurucz-1994 SED for T_{eff}=40kK, Log g=4.5 and Z=$0.02\cdot 10^{-0.5}$=0.0063
1c. `IDL>Sed_D45_002=Read_Paul2000('d45',Z=0.02,Out='Paul00_45_002.sed')`
Obtain the dwarf Pauldrach-2000 SED for T_{eff}=45kK and Z=0.02
1d. `IDL>Sed_S35_001=Read_Paul2000('s35',Z=0.01,Out='Paul00_35_001.sed')`
Obtain the supergiant Pauldrach-2000 SED for T_{eff}=35kK and Z=0.01
1e. `IDL>Sed3030=Read_Paul1997(30000,3.0,Z=0.02,Out='Paul97_30_30_002.sed')`
Obtain the Pauldrach-1997 SED for T_{eff}=30kK, Log g=3.0 and Z=0.02
1f. `IDL>SedB3=Read_Sch('B3',0.004,Out='Sch_B3_004.sed')`
Obtain the SED for the third evolutionary stage of a M_{init}=25$M_\odot$ Schaerer-1996 SED model at Z=0.004
1g. `IDL>Sed96=Read_Stars96('Stars96.output',Out='Stars96.sed')`
Convert a STARS96 output file to MICE struct and MICE file
1h. `IDL>Sed97=Read_Stars97('Stars97.output',Out='Stars97.sed')`
Convert a STARS97 output file to MICE struct and MICE file
1i. `IDL>Sed485=Blackbody(48500,Out='b485.sed')`
Compute a black body SED for T_{eff}=48500K
1j. If the SED of interest is a CLOUDY nebular continuum as contained in a CLI struct, then use the following routines to obtain the MICE SED struct and MICE SED file:
`IDL>Extract_CloudyCont,CLI,'attenuated','Atten.sed'`
`IDL>Sed=Read_Model('Atten.sed')`
2. **Use *FluxFrac* to compute bolometric quantities as well as partial range quantities:**
2a. `IDL>F=FluxFrac(Sed=Sed5038)`
Compute bolometric quantities as well as Lyman continuum ($\lambda \leq 912\AA$) quantities, using a MICE SED struct as input
2b. `IDL>F=FluxFrac(Inname='Kur3540.sed',1,xmin=1E15,xmax=1E16)`
Compute bolometric quantities as well as quantities for the range 10^{15}–10^{16} Hz, using a MICE SED file as input
2b. `IDL>F=FluxFrac(Inname='Stars97.sed',2,xmin=3000,xmax=7000)`
Compute bolometric quantities as well as quantities for the range 3000–7000 $\AA$, using a MICE SED file as input

Display SED shapes

The routine *Plot_Atmos* can be used to display and visualy compare any SED. SEDs to be displayed have to be available either as MICE SED struct or as MICE SED file.

1. **Examples showing how to create MICE SED structs and MICE SED files:**
1a. `IDL>Sed3540=Read_Kur(35000,4.0,Out='Kur3540.sed')`

Obtain the Kurucz-1994 SED for T_{eff}=35kK, Log g=4.0 and Z=0.02

1b. `IDL>Sed4045=Read_Kur(40000,4.5,ZSolar=-0.5,Out='Kur4045.sed')`
Obtain the Kurucz-1994 SED for T_{eff}=40kK, Log g=4.5 and Z=0.02·$10^{-0.5}$=0.0063

1c. `IDL>Sed_D45_002=Read_Paul2000('d45',Z=0.02,Out='Paul00_45_002.sed')`
Obtain the dwarf Pauldrach-2000 SED for T_{eff}=45kK and Z=0.02

1d. `IDL>Sed_S35_001=Read_Paul2000('s35',Z=0.01,Out='Paul00_35_001.sed')`
Obtain the supergiant Pauldrach-2000 SED for T_{eff}=35kK and Z=0.01

1e. `IDL>Sed3030=Read_Paul1997(30000,3.0,Z=0.02,Out='Paul97_30_30_002.sed')`
Obtain the Pauldrach-1997 SED for T_{eff}=30kK, Log g=3.0 and Z=0.02

1f. `IDL>SedB3=Read_Sch('B3',0.004,Out='Sch_B3_004.sed')`
Obtain the SED for the third evolutionary stage of a M_{init}=25$M_\odot$ Schaerer-1996 SED model at Z=0.004

1g. `IDL>Sed96=Read_Stars96('Stars96.output',Out='Stars96.sed')`
Convert a STARS96 output file to MICE struct and MICE file

1h. `IDL>Sed97=Read_Stars97('Stars97.output',Out='Stars97.sed')`
Convert a STARS97 output file to MICE struct and MICE file

1i. `IDL>Sed485=Blackbody(48500,Out='b485.sed')`
Compute a black body SED for T_{eff}=48500K

1j. If the SED of interest is a CLOUDY nebular continuum as contained in a CLI struct, then use the following routines to obtain the MICE SED struct and MICE SED file:
```
IDL>Extract_CloudyCont,CLI,'attenuated','Atten.sed'
IDL>Sed=Read_Model('Atten.sed')
```

2. **Initialize MICE plot window (load color table & plot window definition):**
```
IDL>MiceWin
IDL>MiceCol
```

3. **Use *Plot_Atmos* to compare two SEDs:**

3a. `IDL>Plot_Atmos,Inname='Stars96.sed'`
`IDL>Plot_Atmos,Sed=SedB3,/Over`
The STARS96 model and the blackbody model are plotted in a plot of wavelength ($\AA$) versus specific luminosity (10^{-30} erg/s/$\AA$). The flux units of both SEDs are not compatible: 10^{-30} erg/s/$\AA$ for the STARS96 SED; erg/cm^2/s/$\AA$ for the black body. A warning is issued

3b. `IDL>Plot_Atmos,Inname='Sch_B3_004.sed',XPres=-3,YPres=5`
`IDL>Plot_Atmos,Sed=Sed4045,/Over`
The Schaerer-1996 model and a Kurucz-1994 model are plotted in a plot of energy (linear Rydberg) versus flux density (log Jy)

4. **Add the ionization edges of Helium to the existing plot:**
`IDL>Plot_Atmos,Ele='he'`

Combine a couple of SEDs

In some cases it is useful to be able to combine a couple of SEDs into a new one and use the merged SED as input SED shape for CLOUDY. The routine *Merge_Sed* fulfills this task and allows up to ten SEDs to be combined.
The SEDs to be combined have to be available as MICE SED structs.

1. **Examples showing how to create MICE SED structs:**

1a. `IDL>Sed3540=Read_Kur(35000,4.0)`
Obtain the Kurucz-1994 SED for T_{eff}=35kK, Log g=4.0 and Z=0.02

1b. `IDL>Sed4045=Read_Kur(40000,4.5,ZSolar=-0.5)`
Obtain the Kurucz-1994 SED for T_{eff}=40kK, Log g=4.5 and Z=0.02·$10^{-0.5}$=0.0063

1c. `IDL>Sed_D45_002=Read_Paul2000('d45',Z=0.02,Out='Paul00_45_002.sed')`

Obtain the dwarf Pauldrach-2000 SED for T_{eff}=45kK and Z=0.02

1d. `IDL>Sed_S35_001=Read_Paul2000('s35',Z=0.01,Out='Paul00_35_001.sed')`
Obtain the supergiant Pauldrach-2000 SED for T_{eff}=35kK and Z=0.01

1e. `IDL>Sed3030=Read_Paul1997(30000,3.0,Z=0.02,Out='Paul97_30_30_002.sed')`
Obtain the Pauldrach-1997 SED for T_{eff}=30kK, Log g=3.0 and Z=0.02

1f. `IDL>SedB3=Read_Sch('B3',0.004)`
Obtain the SED for the third evolutionary stage of a M_{init}=25$M_\odot$ Schaerer-1996 SED model at Z=0.004

1g. `IDL>Sld96=Read_Stars96('Stars96.output')`
Convert a STARS96 output file to MICE SLD struct and MICE file

1h. `IDL>Sld97=Read_Stars97('Stars97.output')`
Convert a STARS97 output file to MICE SLD struct and MICE file

1i. `IDL>Sed485=Blackbody(48500)`
Compute a black body SED for T_{eff}=48500K

2. **Initialize MICE plot window (load color table & plot window definition):**
`IDL>MiceWin`
`IDL>MiceCol`

3. **Combine a couple of SEDs into a new one:**

3a. `IDL>Aver=Merge_Sed(Sed3540,Sed_D45_002,[0.5,0.5],OutName='Aver.sed')`
The average SED is computed from two input SEDs. The combined SED is available as MICE SED struct and as MICE SED file

3b. `IDL>Aver=Merge_Sld(Sld96,Sld97,[0.5,0.5],OutName='Aver.sld')`
The average SLD (spectral luminosity distribution) is computed from two input SLDs. The combined SLD is available as MICE SLD struct and as MICE SLD file

3c. `IDL>Sum=Merge_Sed(Sed3540,Sed4045,Sed3=3030,Sed4=Sed_D45_002,[4,2,1,0.5])`
Four SEDs are combined into a new one. The contribution of SED one is multiplied by 4, that of SED 2 by 2, that of SED 3 by 1 and that of SED 4 by 0.5 before addition

How to create and use your own SED file

If you want to compute a CLOUDY model using your own external SED instead of the external SEDs contained in the MICE library, you should undertake the following steps:

1. The first line of your new SED file *my.sed* should contain an SED title. This title must be generated by the *Make_Sed_Title* routine. To generate a title for an SED of T_{eff}=3100 K, log g=4.12, Z=0.04 and R_{star}=4.6, type the following:
`IDL>Title=Make_Sed_Title('USERDEF',Teff=3100,Glog=4.12, Z=0.04,RStar=4.6)`
The contents of `Title` (including the empty space at the beginning of the string) should be put on the first line of the SED file *my.sed*. (If not all four parameters are known or applicable, then leave the respective keyword(s) out)
2. The second line of the SED file should be left blank.
3. Append the SED file with columns of wavelength & flux stated in any of the following units:
Wavelength: [Ryd] [Hz] [um] or [A]
Flux: [erg/cm∧2/s], [erg/cm∧2/s/Hz], [Jy], [erg/cm∧2/s/um], [erg/cm∧2/s/a] or [edd erg/cm∧2/s/Hz] (=Eddington flux)
4. The SED file *my.sed* with wavelength and flux stated in units [μm] and [Jy] can be displayed in MICE by typing:
`IDL>R=Read_Model('my.sed',frequnit='um',fluxunit='jy')`
`IDL>Plot_Atmos(sed=r,xpres=2,ypres=5)`
This plots the SED in logarithmic units [μm] and [Jy]

5a.To be able to use your own external SED in Cloudy, first save your SED in the prefered

MICE wave & flux units as, for instance, *my.sed.mice*:

```
IDL>R=Read_Model('my.sed',frequnit='um',fluxunit='jy',outname='my.sed.mice')
```

5b. Now you can specify your converted own external SED *my.sed.mice* in the MICE input file:

```
External Model (my.sed.mice)
```

The first ten lines of the user defined SED file *my.sed.mice* in the example would look like this:

```
 USERDEF  T =   3100.0  G = 4.12  Z = 0.0400  R =     4.6

9.1134e-04  1.0541e+07
1.5195e-03  2.6333e+07
3.6457e-03  1.5402e+08
7.5942e-03  8.1844e+08
1.8233e-02  7.4516e+09
2.2505e-02  1.3225e+10
2.5672e-02  1.9034e+10
2.6031e-02  1.9803e+10
```