

# USER MANUAL FOR MORE OF DUSTY

Martin Groenewegen\*

Release 8, Januari 2023

## Abstract

MORE OF DUSTY (MOD) is an extension of the well known publically available radiative transfer code DUSTY.

DUSTY is placed as a subroutine within MOD. MOD allows one to determine the best fitting parameters to a set of observations by a Levenberg-Marquardt minimization procedure. These observations can be magnitudes, spectra, intensity profiles or visibility curves.

The fit parameters in the currently released version are luminosity, optical depth, dust temperature at the inner radius, and exponent of the density law. Any of these parameters may also be fixed. Since the source code is made available, users may easily implement other or more fitting parameters.

Input to MOD are a master input file containing first guesses for the fit parameters, the assumed distance, the assumed  $A_V$ , a file with the absorption and scattering coefficients, a stellar atmosphere model and effective temperature, and the density structure. Additional input files contain the observational data, i.e. observed magnitudes, spectra, intensity profiles and/or visibility curves.

The output are the usual output files given by DUSTY for the best fitting model, and the best fitting parameters with errorbars.

For users familiar with the plotting package PGPLOT, additional plots showing a comparison between models and observations may be generated (but this is not part of the standard release of MOD).

This is the manual describing MOD, not DUSTY! Familiarity with the use of DUSTY is assumed and probably required.

---

\*Current address: Royal Observatory of Belgium, Ringlaan 3, B-1180 Brussels, Belgium

This code is copywrited, 2012-2023 by Martin Groenewegen, and may not be copied without acknowledging its origin. Use of this code is not restricted, provided that acknowledgement is made in each publication.

The bibliographic reference is Groenewegen (2012, A&A 543, A36; hereafter G12).

When using MoD also acknowledge the use of DUSTY,

Ivezić, Ž., Nenkova, M. & Elitzur, M., 1999, User Manual for DUSTY, University of Kentucky Internal Report, accessible at <http://www.pa.uky.edu/~moshe/dusty>.

# Contents

<b>1</b>	<b>INTRODUCTION</b>	<b>4</b>
<b>2</b>	<b>INSTALLATION</b>	<b>4</b>
2.1	PGPLOT users . . . . .	5
<b>3</b>	<b>CHANGES TO DUSTY ITSELF</b>	<b>5</b>
<b>4</b>	<b>THE MINIMISATION PART</b>	<b>6</b>
<b>5</b>	<b>INPUT TO MOD</b>	<b>7</b>
<b>6</b>	<b>OUTPUT OF MOD</b>	<b>13</b>
<b>7</b>	<b>Caveats and disclaimer</b>	<b>15</b>
<b>8</b>	<b>Error messages</b>	<b>15</b>
	<b>APPENDICES</b>	<b>17</b>
<b>A</b>	<b>THE USE OF PGPLOT</b>	<b>17</b>
A.1	Error messages . . . . .	17

# 1 INTRODUCTION

The code **DUSTY** was developed at the University of Kentucky by Željko Ivezić, Maia Nenkova and Moshe Elitzur for a commonly encountered astrophysical problem: radiation from some source (star, galactic nucleus, etc.) viewed after processing by a dusty region. The original radiation is scattered, absorbed and reemitted by the dust, and the emerging processed spectrum often provides the only available information about the embedded object.

An equally commonly encountered problem is what set of parameters actually best fit a set of observations. Within the framework of **DUSTY** one could easily run a set of models of varying optical depth and then compare the **DUSTY** output to the observations using additional software. However, if one would also like to vary other parameters, e.g. grain size, dust composition, or effective temperature one would have to run many more models. Under certain conditions it could be advantageous to directly perform a minimisation of these parameters.

This is the aim of MORE OF **DUSTY** (MyDusty): the **DUSTY** code is used as a subroutine within a minimisation routine. The observational data that can be fitted are: photometry, spectroscopy, intensity profiles and visibility curves.

## 2 INSTALLATION

The tar file provided contains this user manual, the official **DUSTY** user manual for easy reference (**DUSTY**manual.ps), two subdirectories for the stars TT Cyg and OH 26.5<sup>1</sup>, and the subdirectory **GRAINS**. Each subdirectory contains two versions of the code, **MyDusty\_v16.f** and **MyDusty\_v16S.f**, for fitting one and two shells, respectively, as outlined in G12. The subdirectory **GRAINS** contains the filter response curves.

The subdirectory **ModelsDust** contains examples of dust species and model atmospheres. This directory is strictly speaking not part of MoD, as these files are external to the code, and you are free to change them. These files are provided so that the examples run correctly on your local computer. Models for carbon-rich stars start with

**mxcom**

and come from Aringer et al. (2009). Models for oxygen-rich start with

**s**

and come from Gustafsson et al. (2008). The number after the *t* or *s* indicates the effective temperature. In both cases the original models are convolved and rebinned to a lower resolution.

MOD is written in FORTRAN 77, and producing the executable file is rather straightforward. For example, on a Unix machine (with pgplot installed) something like

```
gfortran -fbounds-check -O -o MyDusty.exe MyDusty_v16.f -lpgplot
```

---

<sup>1</sup>Update for release 8: another subdirectory for the star LS Pup was created. This is galactic cepheid with some IR excess. A model was presented in the 2023 paper on the modelling of the SEDs of classical cepheids in the Magellanic Clouds.

should work. The code was tested with the f77 and gfortran compilers on a 32-bit Linux machine running Ubuntu 8.04, a 64-bit Linux machine running Ubuntu 14.04 (and later), and with gfortran and ifort on a 64-bit machine running SUSE 10.

The compilation gives some warnings that can safely be ignored.

Update November 2020: with a more recent version of gfortran one “error” was reported that used to be a “warning”<sup>2</sup> Adding the flag “-std=legacy” will turn it back into a warning, and the compilation should be successful.

## 2.1 PGPLOT users

A usefull feature is some plotting facilities using the PGPLOT package<sup>3</sup>. In the compilation of MyDusty you would have to add the appropriate libraries (-lpgplot suffice in my case, but on some other systems I have worked on over time I also had to link -lX11 for example). If you dont have pgplot, dont like it, or dont want to install it, see the Appendix.

In addition there is a separate code, PlotMyDusty, which allows the comparison of the SED and spectra to the best fitting model. PlotMyDusty is also based on PGPLOT, and details are listed in the Appendix.

## 3 CHANGES TO DUSTY ITSELF

The core of MOD is DUSTY version 2.01, and the changes described below are w.r.t. that version

- In the process of developing MOD, one bug was found and corrected, namely in subroutine visi2d where the loop over the radial positions ran from (1,Nin), but should run from to Nin-1.
- Elems(25,npL). The second dimension was increased to Elems(25,npP)<sup>4</sup>.
- In DUSTY it is possible to calculate intensity profiles (the \*.itb files). From an observational point one would prefer intensity profiles convolved with the beam of the instrument/telescope.

In fact, this option was already present in DUSTY (subroutines Convolve and Conv2D) but not described in the DUSTY user manual.

The calculation requires parameters for the convolution (described later when discussing the input), but also the angular diameter of the inner dust shell. In DUSTY this number was completely independent of any observational parameter (Luminosity, effective temperature, dust condensation temperature, which basically set this

---

<sup>2</sup>CALL LINSYS(Nanal,A,yaux,coeff,error)

Error: Actual argument contains too few elements for dummy argument ‘x’ (4/109) at (1) [-Wargument-mismatch]

<sup>3</sup>see <http://www.astro.caltech.edu/~tjp/pgplot/> and it is easily installable via e.g. the Ubuntu packet manager

<sup>4</sup>It is recommended to compile the code including a check on array bounds, e.g. -fbounds-check, which actually led to the discovery of these errors.

value). In MOD the angular diameter of the inner dust shell must be calculated self-consistently. In order to achieve this, the call to the subroutines that calculate the convolved intensity profiles (and visibilities as well) had to be moved to a different location in DUSTY.

The output of the convolution is written to a new file, \*.itbc

The file contains:

- a header line
  - a number indicating the number of wavelengths
  - The list of wavelengths, which is copied a second time for convenience
  - The FWHM extent of the dust shell, determined as the square root of (FWHM of the convolved intensity profile)<sup>2</sup> - (FWHM of the telescope beam)<sup>2</sup>, which is given as input (see later).
  - a dummy line
  - first column the offset in arcsec, followed by the intensity profiles for the wavelengths, first in units of Jy/arcsec<sup>2</sup>, and then in MJy/sr
- In MoD Release 4 a feature was corrected. In previous Releases the convolution was not calculated accurately enough, in particular the angular integration. The parameter Nphi in subroutine Conv2D was increased from 9 to 108.
  - Strictly speaking unrelated to the implementation of MOD, a more general form of the piecewise power law for the density structure (density type = 1) was implemented in DUSTY.

$$\eta(y) \propto \begin{cases} s(1) y^{-p(1)} & 1 \leq y < y(1) \\ s(2) y^{-p(2)} & y(1) \leq y < y(2) \\ s(3) y^{-p(3)} & y(2) \leq y < y(3) \\ \vdots & \\ s(N) y^{-p(N)} & y(N-1) \leq y \leq y(N) \end{cases} \quad (1)$$

With such a density structure it is possible to model steps in the mass-loss rate.

This implementation required modification in subroutine Ygrid in order to better resolve the different regions, and a change in subroutines ETA and Product. The parameter EtaRat, that is a control parameter on how much the normalized density profile may change between two radial gridpoints, was set to a large value.

## 4 THE MINIMISATION PART

The minimisation part is based on the MRQMIN routine (and the routines that are called: mrqcof, gaussj, covsrt) from Press et al (1992, Numerical Recipes).

Although the input parameters are e.g. optical depth, luminosity and temperature at the inner radius, the parameters that are being fitted are the log values. This is to ensure positive numbers for these quantities and also to ensure that all fit parameters are of order unity.

In order to perform the minimisation, the derivatives of  $\chi^2$  to the fit parameters are needed. This is done via a numerical computation,  $(f(x+h) - f(x))/h$ , with  $h = 0.01$  for all parameters.

What is minimized is

$$\chi^2 = \sum_i (m_{\text{obs}}(i) - m_{\text{pred}}(i))^2 / (\sigma_{\text{m}_{\text{obs}}(i)})^2, \quad (2)$$

with  $m$  the observed or predicted quantity (magnitude in a band, spectral flux at a wavelength, convolved intensity at a certain offset, or visibility at a certain spatial frequency) with error bar  $\sigma_{\text{m}_{\text{obs}}}$ . Also the reduced  $\chi^2$  is calculated:

$$\chi_r^2 = \frac{\chi^2}{(n - p)}, \quad (3)$$

with  $n$  the number of data points, and  $p$  the number of free parameters, and the quantity

$$\text{BIC} = \chi^2 + \sum_i \ln(\sigma_{\text{m}_{\text{obs}}(i)})^2 + (p + 1) \ln(n). \quad (4)$$

This is based on the Bayesian information criterion (Schwarz 1978) and measures whether an increase in the number of free parameters and the resulting lower  $\chi^2$  is actually significant.

## 5 INPUT TO MOD

MyDusty requires the following input files

- A master input file of arbitrary name (e.g. `oh26.5.in`)<sup>5</sup>

MOD is then run typing on the command line `MyDusty.exe < filename >`

An example of an input file is:

```
oh26.5
1.5
dhs0.7_a0.20alsil100feo2fe3o41.dat
s2600_g+0.0_m5.0_t02_st_z+0.00_a+0.00_c+0.00_n+0.00_o+0.00_r+0.00_s+0.00.dusty
2600.
1.4
1
1600.
-1.0
1.0
11000. 1
200. 1
700. 0
2.5 1
```

---

<sup>5</sup>As two versions of the code are provided, 16 and 16S for fitting one or two shells, there are also two input files provided `< filename >.in.v16` and `< filename >.in.v16S`.

which signify respectively:

An arbitrary string, but the *root* name of additional input file names<sup>6</sup>

Interstellar reddening,  $A_V$

Name of the file containing the absorption and scattering coefficients (following DUSTY convention)<sup>7</sup>

Name of the file containing the stellar input spectrum (following DUSTY convention)<sup>8</sup>

Effective temperature (K)

Distance in kpc

Then, referring to Eq. 1:

Number of shells,  $N$

Outer radius (radii),  $y(N)$ , of these shells

Exponent of the density law,  $p(N)$ , of these shells. Note that  $p(1)$  is one of the fit parameters in the current version of the code, so its value can be arbitrary (and put to -1 here).

scaling factors,  $s(N)$ .

free parameter 1, luminosity (solar units) and if it is fixed (0), or fitted (1).

free parameter 2, optical depth at 0.55 micron and if it is fixed (0), or fitted (1). As the log value is the parameter that is minimised, **do not put 0 here**, but a small number if you want to start the minimisation from a very small optical depth value.

free parameter 3, temperature at inner dust radius (Kelvin) and if it is fixed (0), or fitted (1).

free parameter 4, slope of the density law,  $p(1)$ , and if it is fixed (0), or fitted (1).

- The file containing the observed magnitudes, *sed2\_root.dat*

An example of this input file is:

esoH	11.270	0.050
esoK	7.530	0.050
2massH	14.580	0.080
2massK	7.980	0.020
iras12	-2.380	0.050
iras25	-4.640	0.050
iras60	-6.270	0.050
ukt800	100.000	10.000
uk1100	48.000	5.000

---

<sup>6</sup>And if the PGLOT option is used the name used as label in the plots

<sup>7</sup>The absorption and scattering coefficients are provided by the user. For easy use the files used in G12 in the case of TT Cyg and OH 26.5 have been provided.

<sup>8</sup>The stellar input spectrum is user input. For easy use the files used in G12 in the case of TT Cyg and OH 26.5 have been provided. Please refer to G12 for proper referencing.



se1300	64.000	8.000
AkaN60	-6.442	0.112
AkarWS	-5.700	0.229
AkN160	-5.651	0.212
pacsr	35400.000	5300.000
pacsb	317000.0	31800.0

The format is (a6,1x, f11.3,1x, f10.3).

MOD computes the magnitudes for a predefined set of filters, and recognises these filters as input. The filter curves come with the MOD distribution (in subdirectory GRAINS). The list of available filters is listed in Table 1 with some reference to the filter curves.

The flux calibration of these filters is based on a zero magnitude for Vega, as described in Groenewegen (2006).

The units are in magnitudes for the Bessell filters up to including the isocam filters. The units for the PACS and SPIRE filters and the other sub-mm bands are in mJy.

**The sloan, GALEX, and Skymapper filters assume magnitudes in the AB-system!**

Reddening is applied following Cardelli et al. (1989) and O'Donnell (1994) for an  $R_V = 3.1$  (subroutine PSCfold).

The file `rted_root.dat` contains the observed flux at earth, in the format wavelength (micron),  $(\lambda F_\lambda)$  (W/m<sup>2</sup>).

- The file containing the observed spectra, `spec_root.dat`

An example of this input file is:

```

2
oh26.5_SWS.dat
2 1.0 1.0
1
2.4 3.3
oh26.5_LWS_MB.dat
2 1.0 1.0
0

```

The first line indicates how many spectra will follow.

Then, for each spectrum, the next line gives the path to the file, and then a line with 3 numbers.

The first number is an integer code which tells MOD the format of the data in the file, the second is a multiplicative factor for all the flux points in the file, the third number a multiplicative factor for all the fluxerror points in the file.

It is possible to exclude certain regions of the spectrum from the fitting. In the example, 4 regions are excluded, with the shortest and longest wavelength indicated. If you want to fit the entire spectrum you still need to put a 0.

Table 1: Available filters, and code to be used in the sed2 file

code	description	code	description	code	description
BesU	Bessell	caspiJ	CASPIR	mips24	Spitzer MIPS
BesB		caspiH		mips70	
BesV		caspiK		mip160	
BesR		caspiL			
BesI					
	OGLE II	saaoJ	SAAO	casw1	ISOCAM SW1
ogleU		saaoH		calw1	ISOCAM LW1
ogleB		saaoK		calw2	ISOCAM LW2
ogleV		saaoL		calw7	ISOCAM LW7
ogleI				calw10	ISOCAM LW10
	Sloan (AB mag)	esoJ	ESO	pacsb	Herschel PACS 70 mu band
sloanu		esoH		pacsg	Herschel PACS 110 mu band
sloang		esoK		pacsr	Herschel PACS 170 mu band
sloanr		esoL			
sloani		esoM		spire1	Herschel SPIRE 250 mu band
sloanz				spire2	Herschel SPIRE 350 mu band
	WFCAM	irac36	Spitzer IRAC	spire3	Herschel SPIRE 500 mu band
wfcamZ		irac45			
wfcamY		irac58			
wfcamJ		irac80		apx870	LABOCA
wfcamH				ap1200	MAMBO
wfcamK		MSXA	MSX band A		
	DENIS			ukt350	JCMT UKT14 instrument
denisI		iras12	IRAS	ukt450	
denisJ		iras25		ukt600	
denisK		iras60		ukt800	
		ira100		ukt850	
				ukt1100	
2massJ	2MASS	esoN	ESO	uk1300	
2massH		esoN1		uk2000	
2massK		esoN2			
		esoN3		se1300	SEST
		esoQ0			
irsfJ	IRSF			AkarS7	Akari MIR
irsfH				Akar9W	
irsfK				AkaS11	
				AkaL15	
				AkL18W	
	Akari IRC	AkaN60	Akari FIS	AkaL24	
AkarN2		AkaWS			
AkarN3		AkarWL			
AkarN4		AkN160			

Table 2: Available filters, continued

code	description	code	description	code	description
vistaZ	VISTA	WISE1	WISE	Hippac	Hipparcos V
vistaY		WISE2		TychoB	Tycho
vistaJ		WISE3		TychoV	
vistaH		WISE4		GALFUV	GALEX (AB mag)
vistaK				GALNUV	
CousR	Cousins	COBE1	COBE-DIRBE	F070W	JWST
CousI		COBE2		F090W	
AF475W	HST	COBE3		F115W	
AF555W		COBE4		F150W	
AF814W		COBE5		F200W	
AF110W		F277W	JWST	F356W	
AF160W		F444W		F770W	
AF127M		F1000W		F1130W	
AF139M		F1280W		F1500W	
AF153M		F1800W		F2100W	
F2550W	JWST	F140M		F158M	
F162M		F182M		F210M	
F250M		F380M		F300M	
F335M		F360M		F410M	
F430M		F460M		F480M	
F560M					
GAIAG	Gaia	GAIABp		GAIARp	
GAIRVS					
ogle4V	OGLE IV	WalV	Walraven	skympu	Skymapper (AB mags)
ogle4I		WalB		skympv	
kepler	Kepler	WalL		skympg	
		WalU		skympr	
		WalW		skympi	
				skympz	

The following formats for the data content of the spectra are implemented, with integer codes:

1, inspired on the Spitzer IRS format, with 4 columns containing, wavelength (micron), flux (Jy), error in the flux (Jy), and spectral order (the last column is not used in the code).

2, with 3 columns containing, wavelength (micron), flux (Jy), error in the flux (Jy)

3, inspired on the IRAS LRS spectral format as generated by the website

[http://isc83.astro.unc.edu/iraslrs/getlrs\\_test.html](http://isc83.astro.unc.edu/iraslrs/getlrs_test.html) , maintained by Kevin Volk. This format contains a dummy line halfway in the file. Units are wavelength (micron), flux ( $\text{W}/\text{m}^2$ ) . An error of 5% of the flux is assumed.

- The file containing the observed intensity profiles, *int\_root.dat*

An example of this input file is:

```
2
OH26.5+0.6_blue_1.00_27.dat
1 69. 5.8 1.0 1 3.2 0.04
OH26.5+0.6_red_2.00_20.dat
1 153. 11.1 1.0 1 6.4 0.07
```

The first line indicates how many intensity profiles will follow.

Then, for each intensity profile, the next line gives the path to the file, and then a line with 6 numbers.

The first is an integer code which tells MOD the format of the data. The second number is the wavelength in micron. The third is the FWHM (in arcsec) of the instrument, to be used in the convolution, and assumed to be a Gaussian. The fourth is a multiplicative factor for the error bars. The fifth is an integer indicating the number of header lines in the datafile. The sixth indicates the pixel size of the instrument (in arcsec).

If the pixel size is a negative number it is not used. If positive, the convolved intensity profile is numerically integrated over a square pixel of the given size, before the normalised intensity profile as a function of offset is calculated.

Currently, only one dataformat is implemented, with integer code:

1, format: offset in arcsec, intensity (arbitrary units but should represent a flux level per surface area, e.g. Jy/pixel, or MJy/sr), dummy third column.

The surface brightness need not to be normalised to unity. This is done inside the code.

The adopted error is the maximum of 10% of the intensity and the rms level (see below).

After having read in the profile, an iterative procedure is used to exclude the typically noisy points further away from the star. By excluding the points brighter than a certain level, 1% of the peak flux in the first iteration, and later on 3 times the rms noise, the rms level is determined in the remainder of the profile. After the iteration is completed, only those points are kept in the fitting until the flux level drops below the rms level for the third time.

- The file containing the observed visibility curves, *vis\_root.dat*

An example of this input file is:

```
5
OH26_Driebe_K.dat
3  2.13  0 1.0 0.99
OH26_visib_CF_5.dat
3  4.9   0 1.0 0.8
OH26_visib_FC_87.dat
3  8.7   0 1.0 0.8
OH26_visib_FC_98.dat
3  9.8   0 1.0 0.8
OH26_visib_CF_10.dat
3 10.0   0 1.0 0.8
```

The first line indicates how many visibility curves will follow.

Then, for each visibility curve, the next line gives the path to the file, and then a line with 5 numbers.

The first is an integer code which tells MOD the format of the data. The second number is the wavelength in micron. The third is an integer indicating the number of header lines in the datafile. The fourth is a multiplicative factor for the error bars. The fifth is interpreted as an angular diameter in arcsec.

This last parameter is related to the field-of-view of the interferometer, which can be smaller than the size of the envelope. In DUSTY the spatial integration of the visibility curve is set to the minimum of the outer radius of the dust shell (which is one of the standard input parameters to DUSTY) and the radius equivalent to the given angular diameter.

The following formats for the data content of the visibility curves are implemented, with integer codes:

- 1, spatial frequencies in units of  $10^5/\text{rad}$ , visibility, error in the visibility.
- 2, spatial frequencies in units of  $\text{arcsec}^{-1}$ , visibility. An error of 0.03 is adopted.
- 3, spatial frequencies in units of  $\text{arcsec}^{-1}$ , visibility, error in the visibility.

## 6 OUTPUT OF MOD

The result of the fitting procedure is saved in the standard DUSTY input file *root.inp* of the best fitting model.

The fit results are at the end of this file:

- The line starting with chi2 lists the overall reduced  $\chi^2$  and BIC, and the number of parameters that were fitted.
- Then follow n-lines for the n-fit parameters, with value, error and if it was fixed (0) or fitted (1).

- The line starting with allchi lists the value  $\chi^2/(\text{number of observations})$  for respectively, the photometry, spectroscopy, intensity profiles, visibility curves.

- Finally, for convenience, the outer radius, the stellar radius and the inner dust radius in units of the stellar radius are listed on separate lines.

-(in Release 4 onwards) the value of  $\log g$  is listed, as well as the (gas) mass-loss rate, assuming a dust-to-gas ratio of 0.005 and an expansion velocity of 10 km/s (both values are hardcoded, look in the subroutine MLrate), and reading the actual grain size and grain specific density from the file provided.

```
Spectrum = 5
s2600_g+0.0_m5.0_t02_st_z+0.00_a+0.00_c+0.00_n+0.00_o+0.00_r+0.00_s+0.00.dusty
optical properties index = 3
dhs0.7_a0.20alsil100feo2fe3o41.dat
temperature =      700.00000
density type = 1
N =              2
shells =      3.2634843871865251      503.26348438718645
p =           2.0000000000000000      2.0000000000000000
scaling =      1.0000000000000000      0.22918008126617884
grid type = 1
lambda0 = 0.55 micron
tau(min) =      208.73547307202767      tau(max) =      208.73547307202767
number of models = 1
accuracy for flux conservation = 0.01
verbosity flag      verbose = 2
properties of emerging spectra; fname.spp = 1
detailed spectra for each model; fname.s### = 1
images at specified wavelengths; fname.i### = -1
number of wavelengths =      7
wvl =   69.0   153.0    2.1    4.9    8.7    9.8   10.0
ipsf = 1
theta1 =      12518.571492780557
psftype = 1
fwhm =   5.80   11.10    0.00    0.00    0.00    0.00    0.00
visibilities at spec. wavelengths fname.v### = 1
radial profiles;      fname.r### = 1
detailed run-time messages;      fname.m### = 1
chi2    40.472654461057445      6781.3864597268857      4
      12518.571492780557      24.425543523613669      1
      208.73547307202767      0.4359274882249621      1
      0.22918008126617884      1.9003231844187951E-003      1
      2.0000000000000000      0.0000000000000000      0
      3.2634843871865251      4.0022239191445319E-002      1
      499.99999999999994      0.0000000000000000      0
      2.0000000000000000      0.0000000000000000      0
allchi   546.2    18.3    59.4    4.4
```

```

outer radius (arcmin)  0.45996888815058945
stellar radius (Rsun & mas)  551.03897465199498      1.8274080411876634
inner radius (Rstar)    30.194096006168117
log g=  -2.33155757134353436E-002  +log(M/Msol)
For: a= 0.15 DTG= 0.50E-02 Vexp= 10.00  grain_density= 1.34 then MasslossRate= 0.411E-1

```

## 7 Caveats and disclaimer

This code is provides *as is*.

Martin A.T. Groenewegen and the Royal Observatory of Belgium can not be held responsible for anything the code can or can not do.

Since the problem of minimisation is complex, it is obviously not guaranteed that the model will converge to a minimum in the first place, especially when the starting values are not appropriate. Some testing and common sense are required.

## 8 Error messages

- "Fortran runtime error: Array reference out of bounds for array 'h', upper bound of dimension 1 exceeded"

Happens typically when of the shells extents to beyond the outer radius. Try with another initial guess, or fix either the inner radius or tickness of the shell.

- In case of any error check the *root\_debug.dat* file. The file starts by listing the photometry that it has read. The 4th column lists the integer (printed as a real) code of the filter. If this is zero than there was a typo or a space wrongly placed in the *sed2\_root.dat* file.

## References

- [ ] Aringer, B., Girardi, L., Nowotny, W., Marigo, P., & Lederer, M. T. 2009, A&A, 503, 913
- [1] Cardelli, J.A., Clayton, G.C., & Mathis, J.S. 1989, ApJ, 345, 245
- [2] O'Donnell J.E. 1994, ApJ, 422, 158
- [3] Groenewegen, M.A.T. 2006, A&A 448, 181
- [4] Groenewegen, M.A.T. 2012, A&A, in press
- [ ] Gustafsson, B., Edvardsson, B., Eriksson, K., et al. 2008, A&A, 486, 951
- [5] Ivezić, Ž. & Elitzur, M. 1995, ApJ, 445, 415
- [6] Ivezić, Ž. & Elitzur, M. 1996, MNRAS 279, 1011
- [7] Ivezić, Ž. & Elitzur, M. 1996, MNRAS 279, 1019
- [8] Ivezić, Ž. & Elitzur, M. 1997, MNRAS 287, 799;  
Erratum: MNRAS 303, 864 (1999)
- [9] Press, W.H., Teukolsky, S.A., Vetterling, W.T., & Flannery, B.P. 1992, in Numerical Recipes in Fortran 77, Cambridge U.P.
- [10] Schwarz, G., 1978, Ann. Stat., 6, 461



# APPENDICES

## A THE USE OF PGPLOT

In order to *not* use the PGPLOT facilities build in the code, you would have to comment the following lines: the calls to `pgbegin`, `pgscf`, `pgsch`, `pgend` near lines 17320 and 18000 of the code. In subroutines `plotvis` and `plotint`, all lines of the form “call pg\*\*\*\*\*”

Currently, the code creates the files `root_int.ps` and `root_vis.ps` which show the best fit to the data. Currently the limit is for 6 panels, but this could easily be changed.

Additionally there is a separate code `PlotMyDusty.f` which allows to compare the observed SED and spectra to the model.

The input file to the code is the same as for MOD, so `PlotMyDusty.exe < filename >`

The appearance of the plot is set by the file `Plotlimits.dat`, which should look like:

```
#set plotlimits for PlotMyDusty.f ; FOUR header lines
# wavelength min max for SED plot + dynamic range (log) for SED y-axis
# number of windows with spectra + min max in wavelength per window
# min max wavelength for scaling model to observations
0.15 3100.
8.9
2
5. 40.
2. 5.
8.5 9.0
```

There are four header lines.

The first line gives the shortest and longest wavelength (in micron) plotted for the SED, which is plotted as  $\log \lambda F_\lambda$  versus  $\log \lambda$ , and where the observed photometry and spectra are plotted.

The maximum along the y-axis of the SED plot is set automatically by the code based on the maximum of the model, the dynamic range (in log units) is set by the user.

Then follows an integer indicating how many additional windows should be plotted (if none, place a 0), and then the shortest and longest wavelength of each window. These additional windows aim at comparing the observed spectra to the model in more detail. In these additional windows, the model is scaled to the observations in order to better compare spectral features. The last line in `Plotlimits.dat` indicates the wavelength interval used to determine this scaling factor.

The output is a file, `root_sed.ps`

### A.1 Error messages

- In case of any error check the `root_debugplot.dat` file. The file starts by listing the photometry that it has read. The 4th column lists the integer (printed as a real) code of the filter. If this is zero than there was a typo or a space wrongly placed in the `sed2_root.dat` file.