BEAGLE User's Guide

BayEsian Analysis of GaLaxy sEds

Jacopo Chevallard
Stephane Charlot

Version Issue: Version 0.1. Last Update: February 6, 2016

Contents

1 Introduction **4**
- 1.1 About This Guide . . . 4
- 1.2 Getting BEAGLE . . . 4
- 1.3 Main features . . . 4

2 Getting started **4**
- 2.1 Preparing to the compilation . . . 4
  - 2.1.1 Mac OS . . . 4
- 2.2 Obtaining the package . . . 5
- 2.3 Compiling the code . . . 5
- 2.4 Setting the environment variables . . . 5
- 2.5 Running the code . . . 7

3 Parameter files **8**
- 3.1 Generic parameters . . . 8
- 3.2 Templates . . . 8
- 3.3 Spectroscopy . . . 8
- 3.4 Photometry . . . 10

4 Data format **13**
- 4.1 Spectroscopic data . . . 13
  - 4.1.1 FITS format . . . 13
  - 4.1.2 ASCII format . . . 13
- 4.2 Photometric data . . . 13
  - 4.2.1 FITS format . . . 13
  - 4.2.2 ASCII format . . . 13
- 4.3 Photometric filters . . . 13

5 Adjustable Parameters **15**
- 5.1 Star formation and chemical enrichment history . . . 15
- 5.2 Nebular emission . . . 16
- 5.3 Dust attenuation . . . 16
- 5.4 Kinematics . . . 17
- 5.5 Instrumental effects . . . 18

6 Code output **19**
- 6.1 `GALAXY PROPERTIES` . . . 19
- 6.2 `STAR FORMATION` . . . 19
- 6.3 `STAR FORMATION BINS` . . . 20
- 6.4 `DUST ATTENUATION` . . . 21
- 6.5 `NEBULAR EMISSION` . . . 21
- 6.6 `MARGINAL SED WL` . . . 22
- 6.7 `MARGINAL SED MASK` . . . 22
- 6.8 `MARGINAL SED` . . . 22
- 6.9 `FULL SED WL` . . . 22
- 6.10 `FULL SED MASK` . . . 22
- 6.11 `FULL SED` . . . 22
- 6.12 `MARGINAL PHOTOMETRY` . . . 22
- 6.13 `ABSOLUTE MAGNITUDES` . . . 23
- 6.14 `APPARENT MAGNITUDES` . . . 23

6.15 POSTERIOR PDF 23

1 Introduction

1.1 About This Guide

1.2 Getting BEAGLE

1.3 Main features

2 Getting started

add cosmological par setting!

In this section, we will explain step by step how to get the RAMSES package and install it, then how to perform a simple test to check the installation.

2.1 Preparing to the compilation

To compile BEAGLE you need the following software to be installed on your machine:

- `CMake`, version ≥ 2.8 (to check the version on your machine type `cmake --version`)
- `GCC`, version ≥ 4.8 (to check the version on your machine type `gcc --version`)
- `Open MPI`

All the above packages are well documented and widely used, so with little effort you should be able to install them on your machine.

2.1.1 Mac OS

On a Mac OS >= 10.9 you should install some `Xcode tools` which are not installed by default. For this it should be enough to run the following command from a shell

```
xcode-select --install
```

Normally, the installation script of BEAGLE takes care of installing all the other dependencies. However, the installation of `OpenBLAS` on a Mac OS may fail. To prevent failures, you may want to install `OpenBLAS` by using `Homebrew` before installing BEAGLE. For this, just type

```
brew tap homebrew/science
brew install homebrew/science/openblas
```

The above command will install OpenBLAS in `/usr/local/opt/openblas`. To allow BEAGLE to find OpenBLAS you can copy (or symlink) the installed files into the BEAGLE installation directory, for instance by typing

```
cp /usr/local/opt/openblas/lib/* install_dir/lib
cp /usr/local/opt/openblas/include/* install_dir/include
```

Once `Homebrew` is installed, you can use it to install the above dependencies with the following commands:

- `CMake`:

  ```
  brew install cmake
  ```

- GCC:

  ```
  brew install gcc --without-multilib --with-all-languages
  ```

- Open MPI:

  ```
  brew install open-mpi --enable-mpi-fortran=all
  ```

2.2 Obtaining the package

2.3 Compiling the code

In this `bin/` directory, you will find a Makefile. The first thing to do is to edit the Makefile and modify the two variables `F90` and `FFLAGS`. Several examples corresponding to different Fortran compilers are given. The default values are:

```
F90 = gfortran
FFLAGS = -x f95-cpp-input $(DEFINES) -DWITHOUTMPI
```

2.4 Setting the environment variables

In order to ease the file I/O we suggest to set the following environment variables. Besides making the code I/O simpler, and keep the different files organised, the advantage of using environment variables is that one can use the same parameter files on different machines, where each machine will have its own environment variables settings.

- `BEAGLE_ROOT`:
- `BEAGLE_PARAM_DIR`:
- `BEAGLE_DATA`: folder containing the data, such as spectra and photometric catalogues, that will be fitted with BEAGLE.
- `BEAGLE_DUST`: folder containing the dust attenuation/extinction curves and dust emission templates.
- `BEAGLE_FILTERS`: folder containing the definition of the transmission curves of the photometric filters, the routine to add new filters, and the filters parameter file associated with a photometric catalogue.
- `BEAGLE_RESULTS`: folder containing the output of BEAGLE.
- `BEAGLE_SF_CHE`: folder containing the star formation and chemical enrichment histories from galaxy formation models, e.g. hydrodynamic simulations and semi-analytic models.
- `BEAGLE_TEMPLATES`: folder containing the stellar population synthesis and photo-ionization templates used in BEAGLE.
- `BEAGLE_TEST_FILES`: folder containing the stellar population synthesis and photo-ionization templates used in BEAGLE.
- `FILTERS`: binary file containing the transmission functions of the photometric filters.

A 'typical' BEAGLE setting would therefore create the folder structure below,

```
root
|----data
     |----HUDF
     |----SDSS
     |----CALIFA
     |----COSMOS
|----dust
     |----attenuation
     |----emission
|----filters
|----results
     |----HUDF
          |----Model_1
          |----Model_2
          |----Model_3
     |----SDSS
          |----Model_1
          |----Model_2
|----sf_che
     |----Eagle
     |----Illustris
     |----deLucia_Blaizot
     |----Durham
|----templates
     |----BC03
     |----M05
     |----FSPS
```

while the environment variables can be set to point to the directories above by adding to the shell configuration file the lines below (for a bash shell, an example for a C-shell is included in the GitHub repository)

```
###################################################
# START BEAGLE environment variables
###################################################
export BEAGLE_ROOT="${HOME}/beagle/files"

export BEAGLE_PARAM_DIR="${BEAGLE_ROOT}/param"
export BEAGLE_TEST_FILES="${BEAGLE_ROOT}/tests"

export BEAGLE_TEMPLATES="${BEAGLE_ROOT}/templates"
export BEAGLE_FILTERS="${BEAGLE_ROOT}/filters"
export BEAGLE_DUST="${BEAGLE_ROOT}/dust"
export BEAGLE_DATA="${BEAGLE_ROOT}/data"
export BEAGLE_SF_CHE="${BEAGLE_ROOT}/sf_che"
export BEAGLE_RESULTS="${BEAGLE_ROOT}/results"

# This environment variable is already defined by BC03 routines, but here you
# redefine it to be sure that the filters used are those in the BEAGLE directory
# tree
export FILTERS="${BEAGLE_FILTERS}/FILTERBIN.RES"
```

```
###################################################
# END BEAGLE environment variables
###################################################
```

2.5 Running the code

3 Parameter files

In this section we illustrate the meaning of the different keywords entering the BEAGLE parameter files.

3.1 Generic parameters

Keyword	Type	Description
VERBOSE	0	No output is printed.
	1	Print only errors.
	2	Print errors and warnings (default).
	3	Print errors, warnings and messages.
SEED	int	Seed for the random number generator. A negative value (default is -1) implies that the seed is chosen automatically based on the system clock.

3.2 Templates

Keyword	Type	Description
TEMPLATES	string	File containing the list of templates SSPs describing stellar emission. (If set the `TEMPLATES NEBULAR` keyword should not be set).
TEMPLATES NEBULAR	string	File containing the list of templates SSPs describing stellar+nebular emission. (If set the `TEMPLATES` keyword should not be set).
TEMPLATES ALPHA ELEMENTS	string	File containing the list of templates SSPs with different $[\alpha/\mathrm{Fe}]$ abundance ratios.
ALPHA DIFFERENTIAL	bool	Whether to include the effect of $[\alpha/\mathrm{Fe}]$ variation in a differential way or not. (It requires the `TEMPLATES ALPHA ELEMENTS` keyword).
SHRINK TEMPLATES WL RANGE	float	Which wavelength range, in Å, for the templates to consider. A negative value means the minimum/maximum allowed range, e.g. -1 10000 will consider the templates from the minimum wavelength and up to 10^4 Å.
REBIN TEMPLATES	float	The width, in Å, of the new templates wavelength bins. This can be used to reduce the number of wavelength points in the templates, to speed the computations that do not require a finer binning.

3.3 Spectroscopy

Keyword	Token	Type	Description

SPECTRUM FILE DESCRIPTION	wl:colNum:	int	Column number (starting from 1) in an ASCII file containing the wavelength array.
	wl:colName:	string	Column name in a FITS table containing the wavelength array.
	flux:colNum:	int	Column number (starting from 1) in an ASCII file, or dimension in a FITS image (starting from 0), containing the flux array. Units must be F_λ.
	flux:colName:	string	Column name in a FITS table containing the flux array.
	fluxerr:colNum:	int	Column number (starting from 1) in an ASCII file, or dimension in a FITS image (starting from 0), containing the flux error array.
	fluxerr:colName:	string	Column name in a FITS table containing the flux error array.
	sky:colNum:	int	Column number (starting from 1) in an ASCII file, or dimension in a FITS image (starting from 1), containing the sky flux array.
	sky:colName:	string	Column name in a FITS table containing the sky flux array.
	mask:colNum:	int	Column number (starting from 1) in an ASCII file, or dimension in a FITS image (starting from 1), containing the mask array.
	mask:colName:	string	Column name in a FITS table containing the mask array.
	flux:conversion:	real	Conversion factor from the units of the flux array to the units adopted in BEAGLE, [$\mathrm{erg\,s^{-1}\,cm^{-2}\,\AA^{-1}}$].
	wl:conversion:	real	Conversion factor from the units of the wavelength array to the units adopted in BEAGLE, [Å].
	wl:dispersion:	string	Dispersion type of the wavelength in a 2D FITS image, it can be '`linear`' (default), '`log`', or '`ln`'.
	wl:type:	string	Type of wavelength, it can be '`air`' (default) or '`vacuum`'.
	redshift:keyword:	string	Keyword indicating the redshift in the header of an ASCII or FITS file (default is '`redshift`').
	min_rel_err:	real	Minimum relative error to add in quadrature to the flux error read from the input file.

MASK OBSERVATION REGIONS	range-#:	2×real	Regions to be masked in the observed spectrum. A negative value in the first element of the region, e.g. $[-1, 5000.]$ means consider the minimum wavelength, while in the second element of the region it means consider the maximum wavelength, e.g. [1.5E+4, -1].
MASK TEMPLATE REGIONS	range-#:	2×real	Regions to be masked in the templates. A negative value in the first element of the region, e.g. $[-1, 5000.]$ means consider the minimum wavelength, while in the second element of the region it means consider the maximum wavelength, e.g. [1.5E+4, -1]. Note that while the `MASK OBSERVATION REGIONS` keyword acts in the observed-frame of the spectrum, this keyword acts in the rest-frame of the templates.
MASK EMISSION LINES		bool	Whether to mask emission lines or not (see Section ?? for details on the adopted algorithm).
MASK BAD SKY		bool	Whether to mask bad sky subtractions not (see Section ?? for details on the adopted algorithm).

3.4 Photometry

Keyword	**Type**	**Description**
FILTERS FILE	string	The `filters file` contains the list of filters used in the photometric catalogue. The format of the filters parameter file is described below.
PHOTOMETRIC CATALOGUE	string	File containing the photometric catalogue.

The `filters file` defines the photometric filters that will be used in BEAGLE, but also the format of the input photometric catalogue.

Token	**Type**	**Description**
object_ID:colNum	int	Column number (starting from 1) in an ASCII file containing the object ID (it is read, and treated, as a `string`).
object_ID:colName	string	Column name in a FITS table containing the object ID.
units	string	Units of the fluxes in the photometric catalogue. It can be either '`Jy`' (default), '`milliJy`', '`microJy`', or '`nanoJy`'.
flux:conversion	real	Conversion factor from the units of the fluxes to the units adopted in BEAGLE, $[\mathrm{erg\,s^{-1}\,cm^{-2}\,Hz^{-1}}]$.
redshift:colNum	int	Column number (starting from 1) in an ASCII file containing the object redshift.

redshift:colName	string	Column name in a FITS table containing the object redshift.
aperture_correction:colNum	int	Column number (starting from 1) in an ASCII file containing the object aperture correction. All fluxes will be multiplied by this factor before being analysed by BEAGLE.
aperture_correction:colName	string	Column name in a FITS table containing the object aperture correction.
index	int	Index of the photometric filter corresponding to the filters defined in the `FILTERS` environment variable, typically the `FILTERBIN.RES` created by bc03 `build_filterbin` command. (One per filter).
flux:colNum	int	Column number (starting from 1) in an ASCII file containing the flux in a given filter. (One per filter).
flux:colName	string	Column name in a FITS table containing the flux in a given filter. (One per filter).
fluxerr:colNum	int	Column number (starting from 1) in an ASCII file containing the flux error in a given filter. (One per filter).
fluxerr:colName	string	Column name in a FITS table containing the flux errir in a given filter. (One per filter).
min_rel_err	real	Minimum relative error to add in quadrature to the flux error read from the input file. (One per filter).

Below we report an example `filters file`, which included different HST bands.

```
# It contains the filters to be used when calculating only photometric models

# If the catalogue contains spec-z you add the following line
# redshift:colName:<redshift column name>
# Example -------> redshift:colName:Z_EAZY

# The possible units that can be used are: Jy, milliJy, microJy, nanoJy
# Alternatively one can pass a conversion factor to convert the fluxes in the
# catalogue into [erg s^-1 cm^-2 Hz^-1] -----> flux:conversion:<conversion factor>
units:microJy

object_ID:colName:ID

index:239              min_rel_err:0.05        flux:colName:FLUX_F225W
↪  fluxerr:colName:FLUXERR_F225W                label:WFC3_UVIS_F225W
index:240              min_rel_err:0.05        flux:colName:FLUX_F275W
↪  fluxerr:colName:FLUXERR_F275W                label:WFC3_UVIS_F275W
index:241              min_rel_err:0.05        flux:colName:FLUX_F336W
↪  fluxerr:colName:FLUXERR_F336W                label:WFC3_UVIS_F336W
```

```
index:219           min_rel_err:0.05        flux:colName:FLUX_F435W
  ↪ fluxerr:colName:FLUXERR_F435W            label:WFC_F435W
index:231           min_rel_err:0.05        flux:colName:FLUX_F606W
  ↪ fluxerr:colName:FLUXERR_F606W            label:WFC_F606W
index:232           min_rel_err:0.05        flux:colName:FLUX_F775W
  ↪ fluxerr:colName:FLUXERR_F775W            label:WFC_F775W
index:234           min_rel_err:0.05        flux:colName:FLUX_F850LP
  ↪ fluxerr:colName:FLUXERR_F850LP           label:WFC_F850LP
index:235           min_rel_err:0.05        flux:colName:FLUX_F105W
  ↪ fluxerr:colName:FLUXERR_F105W            label:WFC3_F105W
index:236           min_rel_err:0.05        flux:colName:FLUX_F125W
  ↪ fluxerr:colName:FLUXERR_F125W            label:WFC3_F125W
index:237           min_rel_err:0.05        flux:colName:FLUX_F140W
  ↪ fluxerr:colName:FLUXERR_F140W            label:WFC3_F140W
index:238           min_rel_err:0.05        flux:colName:FLUX_F160W
  ↪ fluxerr:colName:FLUXERR_F160W            label:WFC3_F160W
```

4 Data format

4.1 Spectroscopic data

BEAGLE assumes that spectroscopic data are expressed in units of F_λ ($\mathrm{erg\,s^{-1}\,cm^{-2}\,\AA^{-1}}$), and the wavelength in units of Å.[1] The units of the input spectrum and of the wavelength can be scaled by a constant factor with respect to such units, and these conversion factors can be passed by using the `conversion` token in the `SPECTRUM FILE DESCRIPTION` keyword of the BEAGLE parameter file (see section 3.3).

4.1.1 FITS format

Spectroscopic data can be supplied in the FITS format, either through FITS tables, or FITS images. In the case of FITS tables, the column names specified in the `SPECTRUM FILE DESCRIPTION` keyword (section 3.3) are used to associate each column of the FITS table to a quantity. In the case of FITS images, the column numbers (starting from 1) specified in the `SPECTRUM FILE DESCRIPTION` keyword define the (second) dimension of the image corresponding to each quantity (the first dimension is assumed to be wavelength). Note that in both cases it is assumed that the table or image are contained in the first (non-empty) extension of the FITS file.

BEAGLE will also search for the `redshift:keyword` in the FITS header to set a redshift for the object.

4.1.2 ASCII format

The header of the ASCII file is ignored, and the different columns numbers (starting from 1) specified in the `SPECTRUM FILE DESCRIPTION` keyword (section 3.3) are used to associate each column of the ASCII file to a quantity.

However, one can specify a redshift for the object by adding the following line to the file header

```
# redshift = real
```

where 'redshift' must match the keyword defined through the `redshift:keyword` token (see section 3.3).

4.2 Photometric data

4.2.1 FITS format

4.2.2 ASCII format

4.3 Photometric filters

To add further filter transmission functions you should modify the `$FILTERS/filterfrm.res` file, by adding **at the end** of the file the new filter transmission function, with the following format $<$ filter $>$

```
# <filter_name>
wl_1 T_1
  .    .
  .    .
  .    .
wl_n T_n
```

[1]To convert a flux density F_ν (in Jy, i.e. $10^{-23}\,\mathrm{erg\,s^{-1}\,cm^{-2}\,Hz^{-1}}$) to F_λ ($\mathrm{erg\,s^{-1}\,cm^{-2}\,\AA^{-1}}$), one can adopt the following expression $F_\lambda = F_\nu\,10^{-23}\,\frac{c}{\lambda^2}$, where both the speed of light c and wavelength λ are expressed in Å.

where `wl_1` to `wl_n` indicate the wavelength (in Å), and `T_1` to `T_n` the corresponding transmission factor.

Once you have modified the `filterfrm.res` file, you must create the corresponding `FILTERBIN.RES` binary file (which is read by BEAGLE). For this, while inside the `FILTERS` folder, run the `build_filterbin` command. To be sure that the newly defined filter(s) has been included, check the `filters.log` file: at the end of the file you should see the new filters.

5 Adjustable Parameters

5.1 Star formation and chemical enrichment history

Name(=default value)	Unit	Description
sfh_type		Type of SF and ChE history to adopt. This can be either • 'ssp'; • 'constant'; • 'exponential', $\psi(t) \propto \exp(-t/\tau)$; • 'delayed', $\psi(t) \propto t\,\exp(-t/\tau)$; • 'rising', $\psi(t) \propto \exp(t/\tau)$. For the above SF and ChE history types, the chemical enrichment is approximated by a single 'metallicity' parameter, i.e. constant with time. For the SF and ChE history below, the metallicity is time-dependent. • 'user', i.e. a user-defined SF and ChE history read from an input ASCII file (see details below); • 'simulated', i.e. based on a set of SF and ChE history obtained from a galaxy formation model (e.g. 'phenomenological' model, semi-analytic model, hydro-dynamic simulation, see details below).
mass	$\log(\mathrm{M}/\mathrm{M}_\odot)$	Mass of stars formed during a star formation period. Note that when using the specific_sfr or sfr parameters, this does not include the mass formed in the 'current star formation' period. The total mass formed in the galaxy is printed in the M_tot column of the GALAXY PROPERTIES extension, while the stellar mass (accounting for the return fraction) in the M_star column of the same extension.
ssp_age	yr	Age of stars when approximating a galaxy by a simple stellar population 'ssp'.
tau	$\log(\tau/\mathrm{yr})$	Star formation timescale during a star formation period (for analytic star formation histories, such as 'exponential', 'delayed' or 'rising').
metallicity	$\log(Z/Z_\odot)$	Metallicity of stars formed during a star formation period described by an 'ssp', 'constant', 'exponential', 'delayed' or 'rising' function.
sfr	$\log(\psi/\mathrm{M}_\odot\,\mathrm{yr}^{-1})$	Star formation rate averaged over the last current_sfr_timescale yr, over which it is assumed constant.

`specific_sfr`	$\log(\psi_S/\mathrm{yr}^{-1})$	Specific star formation rate, computed assuming constant star formation over the last `current_sfr_timescale` yr. Note that this is calculated considering the entire stellar mass formed in the galaxy star formation history, not only the mass actually locked into stars, i.e. it does not account for the return fraction.
`current_sfr_timescale=7`	$\log \mathrm{yr}$	Duration of the current episode of star formation. The default value (10^7 yr) is more appropriate when interpreting emission lines, as these are mainly produced by short-lived stars embedded in their birth clouds, which disrupts on timescales of $\sim 10^7$ yr. When modelling UV data, one may test longer values, up to 10^8 yr, to better account for the longer evolutionary timescale of stars producing the bulk of UV emission in star forming galaxies.
`young_stars_metallicity`	$\log(Z_{\mathrm{young}}/Z_\odot)$	Metallicity of stars younger than 10^7 yr. If not set, and if nebular emission is included in the model, then the metallicity of young stars is set equal to the gas metallicity Z_{gas}, otherwise is set equal to the metallicity of the stars in the most recent star formation period.
`start_age`	yr	Start look-back time of a star formation period.
`end_age`	yr	End look-back time of a star formation period.

5.2 Nebular emission

Name(=default value)	**Unit**	**Description**
`nebular_logU`	$\log \hat{U}$	Effective galaxy-wide ionization parameter.
`nebular_Z`	$\log(Z_{\mathrm{gas}}/Z_\odot)$	Effective galaxy-wide gas metallicity.
`nebular_xi`	ξ_{d}	Effective galaxy-wide dust-to-metal mass ratio.

5.3 Dust attenuation

Name(=default value)	**Unit**	**Description**

attenuation_type		Type of dust attenuation prescription to adopt. This can be either • '`calzetti`', the ? empirical attenuation curve obtained from a sample of 39 nearby starburst and blue-compact galaxies; • '`cf00`', the 2-component model (diffuse ISM + stellar birth clouds) of ?; • '`ccww13_universal`', the 'quasi-universal' relation of ?; • '`ccww13_full`', the 'full model' of ?, based on the radiative transfer calculations of ?.
tauV_eff	$\hat{\tau}_V$	V-band attenuation optical depth ($\hat{\tau}_V$). Parameter required for any `attenuation_type`.
mu=0.3	μ	Fraction of attenuation optical depth arising from the diffuse ISM, i.e. $\hat{\tau}_V^{\mathrm{ISM}} = \mu\,\hat{\tau}_V$, while $\hat{\tau}_V^{\mathrm{BC}} = (1-\mu)\,\hat{\tau}_V$. Parameter required for `attenuation_type=cf00, ccww13_universal, ccww13_full`.
inclination	θ/deg	Galaxy inclination in the dust radiative transfer calculations of ?.
tau_b_perp	$\hat{\tau}_{B,\perp}$	Central face-on B-band optical depth in the dust radiative transfer calculations of ?.

5.4 Kinematics

Name(=default value)	Unit	Description
sigma	$\sigma/\mathrm{km\,s^{-1}}$	Stellar velocity dispersion.
v_sys	$v_{sys}/\mathrm{km\,s^{-1}}$	Stellar systemic velocity.
h3	h_3	Coefficient of the 3rd order Hermite polynomial, to describe the stellar LOSVD.
h4	h_4	Coefficient of the 4th order Hermite polynomial, to describe the stellar LOSVD.
nebular_sigma		Gas velocity dispersion.
nebular_v_sys		Gas systemic velocity.
nebular_h3		Coefficient of the 3rd order Hermite polynomial, to describe gas LOSVD.

nebular_h4		Coefficient of the 4th order Hermite polynomial, to describe gas LOSVD.

5.5 Instrumental effects

Name(=default value)	**Unit**	**Description**
lsf_type		Type of function describing the instrumental line-spread function: • box, a 'box' function; • gaussian, a 'gaussian' function, $\mathcal{N}(\mu_{\mathrm{LSF}}, \sigma_{\mathrm{LSF}})$;
lsf_box_width	Å	Width of the box defining a box LSF.
lsf_sigma	σ_{LSF}/Å	Width of the Gaussian defining a gaussian LSF.
lsf_center	μ_{LSF}/Å	Center of the Gaussian defining a gaussian LSF.
lsf_sigma_coeff-#		Coefficient of the polynomial defining the variation of σ_{LSF} with wavelength. For a polynomial of degree N, one has to define $N+1$ parameters lsf_sigma_coeff-1, lsf_sigma_coeff-(N+1).
lsf_center_coeff-#		Coefficient of the polynomial defining the variation of μ_{LSF} with wavelength. For a polynomial of degree N, one has to define $N+1$ parameters lsf_center_coeff-1, lsf_center_coeff-(N+1).

6 Code output

6.1 GALAXY PROPERTIES

Column name	Units	Description
redshift		Redshift of the galaxy.
sigma	$\mathrm{km\,s^{-1}}$	Stellar velocity dispersion.
M_tot	$\mathrm{M_\odot}$	Total mass of stars formed in the galaxy.
M_star	$\mathrm{M_\odot}$	Total mass locked into stars (it accounts for the fraction of mass returned by stars to the ISM).
mass_w_age	yr	Mass-weighted age, computed considering the mass locked into stars at each time step.
lumin_w_age	yr	Luminosity-weighted age.
mass_w_Z		Mass-weighted metallicity, computed considering the mass locked into stars at each time step.
lumin_w_Z		Luminosity-weighted metallicity.
N_ion	$\mathrm{s^{-1}}/3.826\,10^{33}$	Number of ionizing photons emitted by the galaxy.
xi_ion	$\mathrm{erg\,Hz^{-1}}$	Ionizing emissivity.
UV_slope		Slope of the UV continuum computed from the GALEX FUV and NUV filters.

6.2 STAR FORMATION

Column name	Units	Description
redshift		This is equal to redshift column in the GALAXY PROPERTIES extension.
formation_redshift		The formation redshift of the galaxy, which defines the age of the oldest stars.
SFR	$\mathrm{M_\odot\,yr^{-1}}$	Star formation rate averaged over the last current_sfr_timescale yr (typically 10^7 yr).
sSFR	$\log\mathrm{yr^{-1}}$	Specific star formation rate, computed considering the mass formed in each time step, i.e. without accounting for the return fraction.

max_stellar_age

6.3 STAR FORMATION BINS

All columns reported below are those printed in the case of a single star formation period. When several SF periods are present, then a suffix is appended to indicate the columns pertaining to each period. In the case of two SF periods, the bin_sf_type will be indicate as bin_sf_type-1 and bin_sf_type-2 for the first and second period, respectively. The same suffix is applied to all other columns.

The number of bin_chem_abund columns depends on the number of chemical parameters in the model. In the simplest (and most common) case of a single chemical parameter, the 'metallicity' (which includes all elements heavier than helium), the table will contain the column bin_chem_abund_1 (as indicate in the table below). When considering, for instance, both metallicity and α-elements to iron ratio, then thee will be indicate by bin_chem_abund_1 (metallicity) and bin_chem_abund_2 (α-to-iron).

Column name	Units	Description
bin_sf_type		Indicates the type of star formation history adopted in the bin, translating into a number the different alternatives indicated in Table 6: • $1 \rightarrow$ ssp; • $2 \rightarrow$ constant; • $3 \rightarrow$ exponential; • $4 \rightarrow$ delayed; • $5 \rightarrow$ rising; • $-2 \rightarrow$ user; • $-1 \rightarrow$ simulated.
bin_start_age	yr	Start look-back time of the star formation period.
bin_end_age	yr	End look-back time of the star formation period.
bin_mass	$M_\odot$	Total stellar mass formed during the star formation period (it does not account for the fraction of mass returned to the ISM).
bin_tau	yr	Star formation timescale during the star formation period, in the case of an exponential, delayed or rising star formation history.
bin_ssp_age	yr	Age of stars when describing a star formation period as a simple stellar population.
bin_chem_abund_1		Chemical abundance of stars in the star formation period, for analytic star formation histories.

6.4 DUST ATTENUATION

Column name	Units	Description
attenuation_type		Indicates the type of dust attenuation prescription adopted, translating into a number the different alternatives indicated in Table 8: • 1 → cf00; • 2 → calzetti; • 3 → ccww13_universal; • 4 → ccww13_full.
tauV_eff		V-band attenuation optical depth.
mu		Fraction of attenuation optical depth arising from the diffuse ISM.
tauB_perp		Central face-on B-band optical depth in the dust radiative transfer calculations of ?.
tauV_eff_ang_aver		Angle-averaged V-band attenuation optical depth.
t_birth_clouds	yr	Timescale of disruption of stellar birth clouds.
inclination	deg	Galaxy inclination.

6.5 NEBULAR EMISSION

Column name	Units	Description
logU		Effective galaxy-wide ionization parameter.
Z		Effective galaxy-wide gas-phase metallicity.
xi		Effective galaxy-wide ionization parameter dust-to-metal mass ratio.
line_name@line_wl	$L_\odot$	Luminosity of the line_name line.
line_name@line_wl_EW		Equivalent width of the line_name line.

6.6 `MARGINAL SED WL`

Column name	Units	Description
`wl`	Å	Wavelength array corresponding to the SEDs contained in the `MARGINAL SED` extension.

6.7 `MARGINAL SED MASK`

Column name	Units	Description
`mask`		Mask array corresponding to the SEDs contained in the `MARGINAL SED` extension. Values of 0 correspond to 'false' values, while values of 1 to 'true' ones.

6.8 `MARGINAL SED`

FITS 2D image containing the flux (units of $\mathrm{erg\,s^{-1}\,cm^{-2}\,\AA^{-1}}$) received by an observes at a distance $d_L(z)$ from the galaxy. If the model is computed at $z = 0$, then $d_L = 10\,\mathrm{pc}$. corresponding to each set of parameters printed in the `POSTERIOR PDF` extension. Each column in the 2D image corresponds to a wavelength (printed in the `MARGINAL SED WL` extension), while each row corresponds to a different set of model parameters (printed in the `POSTERIOR PDF` extension).

6.9 `FULL SED WL`

Column name	Units	Description
`wl`	Å	Wavelength array corresponding to the SEDs contained in the `FULL SED` extension.

6.10 `FULL SED MASK`

Column name	Units	Description
`mask`		Mask array corresponding to the SEDs contained in the `FULL SED` extension. Values of 0 correspond to 'false' values, while values of 1 to 'true' ones.

6.11 `FULL SED`

FITS 2D image containing the predicted luminosity (units of $\mathrm{L_\odot\,\AA^{-1}}$, where $\mathrm{L_\odot} = 3.826\,10^{33}\,\mathrm{erg\,s^{-1}}$) corresponding to each set of parameters printed in the `POSTERIOR PDF` extension. Each column in the 2D image corresponds to a wavelength (printed in the `FULL SED WL` extension), while each row corresponds to a different set of model parameters (printed in the `POSTERIOR PDF` extension).

6.12 `MARGINAL PHOTOMETRY`

Flux (units of $\mathrm{erg\,s^{-1}\,cm^{-2}\,Hz^{-1}}$) received by an observer located at a distance $d_L(z)$ from the source.

6.13 ABSOLUTE MAGNITUDES

Magnitudes (AB system) computed from the flux received by an observer located at a distance of 10 pc from the source.

6.14 APPARENT MAGNITUDES

Magnitudes (AB system) computed from the flux received by an observer located at a distance $d_L(z)$ from the source.

6.15 POSTERIOR PDF

Column name	Units	Description
`probability`		Posterior probability.
`likelihood`		Value of the likelihood function.
`param_name`		Value of the different model free parameters.