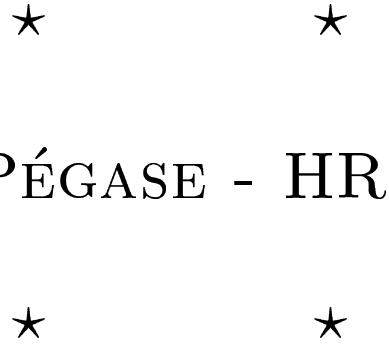

PÉGASE - HR

Projet d'**É**tude des **GA**laxies par **S**ynthèse **É**volutive - **H**aute **R**ésolution

Version 1.1: January 19, 2005

Damien LE BORGNE — Michel FIOC — Brigitte ROCCA-VOLMERANGE
Philippe PRUGNIEL — Ariane LANÇON — Caroline SOUBIRAN

Institut d'astrophysique de Paris
98 bis boulevard Arago, 75014 Paris, France

E-mail: leborgne@iap.fr, fioc@iap.fr

WWW: http://www.iap.fr/pegase/

Contents

1 Introduction

PÉGASE-HR (presented in Le Borgne et al. 2004) is a code aimed at computing synthetic evolutionary optical spectra of galaxies with a very high resolution (R=10 000, or $\delta\lambda = 0.55$ Å) over the wavelength range $\lambda = [4000, 6800]$ Å. It's the result of combining the code PÉGASE.2 with the high-resolution stellar library ÉLODIE.
This code can also be used at low resolution ($R \simeq 200$) over the range covered by the BaSeL library (from far UV to the near IR), and then produces the same results as PÉGASE.2.

The PÉGASE.2 code (Version 2 of PÉGASE, Projet d'Etude des GAlaxies par Synthese Evolutive in French) is aimed at modeling the spectral evolution of galaxies. PÉGASE.2 is presented in Fioc & Rocca-Volmerange (1997) and the code is fully described in a README file (Fioc & Rocca-Volmerange 2000) (`ftp://ftp.iap.fr/pub/from_users/pegase/PEGASE.2/`, also available in the `doc/` directory of this package). PÉGASE.2 uses the BaSeL (Lejeune et al. 1997,1998) library of stellar spectra and can therefore synthesize low-resolution (with a resolution $R \simeq 200$, from 91 Å to 160 μm) ultraviolet to near-infrared spectra of Hubble sequence galaxies as well as of starbursts.

In PÉGASE-HR, the BaSeL library can be replaced by a grid of spectra interpolated from the high-resolution ÉLODIE library of stellar spectra.

The ÉLODIE library is a stellar database of 1959 spectra for 1503 stars, observed with the echelle spectrograph ÉLODIE on the 193 cm telescope at the Observatoire de Haute Provence. Previous versions of the library are presented in Prugniel & Soubiran (2001) and Soubiran et al. (1998); Katz et al. (1998). It has been updated for the present work by doubling the number of spectra, which greatly improved the coverage of the parameter space (in effective temperature, surface gravity, and metallicity).

The data reduction has also been improved, in particular the flux calibration, and the wavelength range has been extended to 400-680 nm. For the purpose of population synthesis, the original resolution R=42 000 has been reduced to R=10 000 at lambda=550 nm, or more precisely to a gaussian instrumental profile of FWHM$\simeq$ 0.55 Å over the whole range of wavelengths. The typical signal-to-noise (S/N) ratio of the spectra is 500/Å. The HR diagram coverage with the estimated stellar parameters is extensive (see Le Borgne et al. 2004 for details).

The up-to-date version of the library of stellar spectra is available at
`http://www.obs.u-bordeaux1.fr/public/astro/CSO/elodie_library.html` where fully reduced spectra, as well as the estimated stellar parameters, are provided.

From this library, a grid of stellar spectra with regularly spaced stellar parameters (the grid has the same underlying shape as the Lejeune et al. grid) was created by interpolation. This grid (`stellibELODIE.fits`), available at `http://www.iap.fr/pegase/`, is used by PÉGASE-HR to compute synthetic spectra.

This document describes the installation procedure, the main changes between PEGASE.2 and PEGASE-HR's algorithms and output formats, and a simplified description of the steps one should follow to compute high resolution synthetic spectra.

2 Installation

2.1 Structure of the package

Before the installation, the package contains the following files:

```
./configure
./src/                          ! Source code
./doc/                          ! Documentation
./data/                         ! Data files
   ./data/tracks/                  -> stellar tracks
   ./data/stellibs/                -> stellar libraries
                                      (stellibLCBcor.fits and stellibELODIE.fits)
   ./data/external/                -> various data files
   ./data/tests/                   -> files used to test the installation
   ./data/user_defined/            -> data files that can be modified by
                                      the user: ages.dat, IMFs, etc.
   ./data/isochrones/              -> pre-computed isochrones (with SSPs_HR.f)
```

If the ÉLODIE library `data/stellibs/stellibELODIE.fits` is not in the package, you should download it from `http://www.iap.fr/pegase/` and copy it into the directory `data/stellibs/`. Warning: the size of this file is 150 Mo.

After the installation, the additionnal directories should be present:

```
./bin/                          ! Binary files
./lib/                          ! Binary libraries
```

2.2 Basic installation

The compilation and installation is made by the standard linux commands:

```
> ./configure
> make
```

Then:

```
  > source peg_cshrc       (if your shell is csh or tcsh)
OR
  > . peg_bashrc           (if your shell is bash)
```

Details of the installation:

- **configure** : `./configure` generates the file Makefile. The presence of the required `cfitsio` library is checked during this process. If this library is not installed on your computer, you can download it from `http://heasarc.gsfc.nasa.gov/docs/software/fitsio/fitsio.html` and install it. If the library is installed but not found, please use the argument "`--with-libraries=`" in your call to `./configure` to specify the path where the library can be found.
  Example: `> ./configure --with-libraries=/usr/lib/`

- **make** : `make` uses the information in `Makefile` to compile the various parts of the PÉGASE-HR code.
  The following binary files are built:

  - `bin/SSPs_HR`
  - `bin/scenarios_HR`

- `bin/spectra_HR`
- `bin/lick`
- `bin/fitstodat`

- Setting up the PATH variable: Depending on the shell you use, you can use the scripts peg_bashrc and peg_cshrc which are created in the "`make`" process. These scripts add the `bin/` directory to the environment variable `PATH`.

2.3 Test of the installation

Once the package is installed, and once you have downloaded the ÉLODIE grid `stellibELODIE.fits` and copied it into the `data/stellib/` directory, you can test the results of your `spectra_HR` compiled code by running

```
> make test
```

This will compute three scenarios and measure Lick indices on a high resolution spectrum, in the temporary directory `test_tmp`. Then it will print the result of the comparison of pre-computed spectra (stored in `data/tests/`) with the spectra computed on your machine.

2.4 Organization of the files

Unlike PÉGASE.2, it is not required to run the PÉGASE-HR codes from the directory containing the data files. During the installation, the codes are compiled in such a way that the data files are always read in the `data/` directory.

The `data/` directory is divided into several sub-directories (see the `data/README` file for more details). The commonly used data files that the user can modify are in the `data/user_defined` directory. The most important files that can be modified (but should not be deleted) are:

```
ages.dat            ! list of the output ages for the spectra (in Myr)
filters.dat         ! definition of the filters' transmission curves
index_table.dat     ! definition of the Lick indices
list_IMFs.dat       ! list of the IMFs to choose from (when using SSPs_HR)
list_tracks.dat     ! list of the stellar tracks to use (when calling SSPs_HR)
list_stellibs.dat   ! list of the stellar libraries to use
                                        (used by SSPs_HR and spectra_HR)
```

2.5 System requirements

- Memory : To run the code `spectra_HR`, you need at least 200 Mo of memory. If necessary, you might want to loosen the system limits by running the following commands:

  ```
  > unlimit
  > limit coredumpsize 0
  ```

- Disk space : you will need a minimum of 70 Mo of space disk for package. This does not include the ÉLODIE stellar library (`stellibELODIE.fits`) which, alone, is 150 Mo large.

3 Differences between PÉGASE.2 and PÉGASE-HR

This section will be useful to users already familiar with PÉGASE.2, and may be skipped by others readers.

3.1 Stellar libraries

PÉGASE-HR is designed to run with a variety of stellar spectral libraries, and two libraries are provided within the distributed package. The use of a FITS format for these libraries speeds up access to the data considerably compared to the previous ASCII format of PÉGASE.2.

The header keyword GRID_TYP defines the grid for which stellar spectra are available (a grid point is characterized by its effective temperature, surface gravity and metallicity). GRID_TYP takes the value STANDARD for the library of Lejeune et al. 1998, in the format adopted for PÉGASE.2 (`stellibLCBcor.fits`). It also takes the value STANDARD for any other library provided on the same grid.
The ÉLODIE grid (`stellibELODIE.fits`) contains fewer spectra than the BaSeL one (Lejeune et al. 1998). For this grid, GRID_TYP takes the value TRUNCATED. The output of `SSPs_HR` will be identical for all stellar libraries that have the same parameter grid (and thus the same GRID_TYP).

3.2 Algorithms

Although the program sources have been reorganized, most algorithms of PÉGASE-HR are identical to those of PÉGASE.2. The sequence of programs to run in order to produce a database of single stellar populations, define a star formation scenario, and compute the corresponding galaxy spectrum, is also unchanged.

PÉGASE.2	PÉGASE-HR
SSPs.f	SSPs_HR.f
scenarios.f	scenarios_HR.f
spectra.f	spectra_HR.f

When used with the low resolution stellar library BaSeL (stellibLCBcor), PÉGASE-HR produces the same synthetic spectra as PÉGASE.2, though in a different format (cf. Sect. 3.4).

The ÉLODIE library being less extended (in metallicity, effective temperature and surface gravity) than the BaSeL library, the algorithm of `SSP_HR.f` had to be modified from PÉGASE.2's version. In `SSPs.f`, the interpolation of the stellar spectra made use of the spectra with metallicities immediately greater and smaller than the metallicity of the tracks. In `SSPs_HR.f`, when no spectra with metallicities close to the metallicity of the tracks are available in the ÉLODIE stellar library, some spectra with metallicities larger or smaller are used in the interpolation.

A new code (`lick.f`) can be used to measure some Lick indices on the evolutionary spectra (see Sect. 4.4 for details.)

3.3 Changes in SSPs files and scenarios files

The use of stellar libraries with different grids makes it necessary to secure the use of SSPs files and of stellar libraries. This is done by the keyword GRID_TYP, as described above.

The files *prefix*`_SSPs.dat` now contain a header specifying the type and the name of the grid used to compute the isochrones. This header is produced automatically by the code `SSPs_HR`. The individual isochrone

files (e.g. *prefix*_tracksZ0.02.dat) now also contain in their header an additionnal line specifying the type of the grid used.

The code scenarios_HR now adds a third line in the header of the scenario files, with the name of the stellar library that will be used to compute the synthetic spectra.

The code spectra_HR then checks the compatibility between the grid used to compute the isochrones (the files prefix_SSP_tracks*.dat) and the grid defined in the scenario, used to compute the synthetic spectrum.

3.4 FITS outputs

To reduce the size of the output files, and to increase the speed of the access to the models, the outputs of the code spectra_HR are now in binary FITS format.

The structure of the output FITS files is described here. The FITS files are made of several blocks (or extensions, or HDU for Header Data Units).

- the primary extension contains, in a binary table, the evolutionary spectra. The first dimension is the number of wavelengths; the second dimension is the number of timesteps (hereafter ntimes). **The spectra are still normalized to one solar mass of galaxy, but they are now in units of $\mathbf{L_\odot.\AA^{-1}}$** (with $L_\odot = 3.826 \times 10^{33}$ erg.s^{-1}), instead of erg.s^{-1}.Å^{-1} in PÉGASE.2.
- the extension ETS_LINES contains the emission lines computed for a typical HII region, following the prescriptions used in PÉGASE.2. It has 2 columns and 61 rows (61 being the number of emission lines computed). The first column contains the wavelengths of the lines. The second column contains, at each row, ntimes values, i.e. the evolution of the flux for each emission line.
- the extension ETS_PARA contains the evolution of all the parameters for the galaxy (masses, metallicities, extinction, rates of supernovae Ia and II, number of Lyman photons, etc.) as described in PÉGASE.2's README file.
- the extension ETS_CONT_WCA, if present, contains a vector of the wavelengths for which the synthetic spectra are given. This extension is omitted if the wavelength scale is linearly regular (which is the case for the ÉLODIE library). In this case, the wavelength scale is fully described in the header of the primary extension by the reference wavelength (keyword CRVAL1) and the wavelength step (keyword CDELT1).
- the extension ETS_LICK, if present, contains the values of the Lick indices (as defined in the file data/user_defined/index_table.dat) measured on the synthetic evolutionary spectra. The first dimension of the binary table is the number of indices; the second dimension is the number of timesteps. This extension is added to the FITS file by the code lick.f.

The code fitstodat included in the package makes it possible to convert a PÉGASE-HR FITS output file to an ASCII file, in the same format as PÉGASE.2's outputs. This code can be useful for people used to PÉGASE.2's outputs and who do not want to change the reading format of their own routines. However, we stress that the ASCII files for PÉGASE-HR outputs are about ten times as large as PÉGASE.2's output files (i.e. approximately 10 Mo, depending on the number of timesteps). Note also that the information on Lick indices, if present, is lost during the conversion.

4 Usage of PÉGASE-HR

Computing spectra with PÉGASE-HR is very similar to the way to do it with PÉGASE.2. If you are not familiar with PÉGASE.2, we advise you strongly to read the PÉGASE.2's README file (included in the `doc/` directory). We describe here the simple steps one should follow to compute a synthetic spectrum with PÉGASE-HR.

4.1 Computing isochrones: SSPs_HR.f

The first step is to compute isochrones for various metallicities. Run

```
> SSPs_HR
```

and make your choices of IMF, model for SNe ejecta, stellar winds, and stellar library. PÉGASE.2's README file describes the way to add other IMFs.

The stellar libraries available by default are:

- `stellibLCB.fits` : Lejeune et al. (1998) low-resolution stellar library
- `stellibELODIE.fits` : ÉLODIE (high-resolution) stellar library interpolated onto a regular grid of stellar parameters.

A file (`prefix_SSPs.dat`) is then created, and contains the list of the isochrone files for every metallicities. The isochrone files (`prefix_tracksZ0.0001.dat` ... `prefix_tracksZ0.1.dat`) are also created in the current directory.

The metallicities of the isochrones computed with the ÉLODIE library can, in principle, be the same as the metallicities of the PÉGASE.2 isochrones. However, the small number of stars with extreme metallicities in this library makes the isochrones with very low or very large metallicities more uncertain.

4.2 Describing scenarios of evolution: scenarios_HR.f

You can now define one or several scenarii of evolution, from the simplest (single instantaneous burst) to the most complex (with infall, galactic winds, continuous star formation, evolution of the metallicity, etc.). Run

```
> scenarios_HR
```

and choose

- the name of the text file created by this code (the scenario itself)
- the name of the file containing the list of isochrones (`prefix_SSPs.dat`). A default name is given, and corresponds to a set of isochrones computed with a Salpeter IMF, stellar winds, and the ÉLODIE library. In the next step (`spectra_HR`), the isochrones will be read automatically in the same directory as `prefix_SSPs.dat`.
- the fraction of sub-stellar objects
- the stellar library you want to use. **Warning: the stellar library should be the same as the one used to create `prefix_SSPs.dat`. If it is not the case, an error will appear in the next step (`spectra_HR`) and no synthetic spectrum will be computed.**

- for each scenario of evolution, the parameters of star formation, as described in details in PÉGASE.2's README.

4.3 Computing synthetic spectra: spectra_HR.f

Once a scenario is defined (and stored in a text file, for example `myscenario.scn`), run `spectra_HR` to compute the evolutionary synthetic spectra:

```
> spectra_HR myscenario.scn
```

If you omit the command-line argument (here, `myscenario.scn`), you will be asked for the name of the file describing the scenarios.

One or several output files (in FITS format, with the structure described in section 3.4) are then created. If these files already exist, the program will abort, issuing an error message.

Important notice: if a file `ages.dat` is present in the current directory, then the output spectra are computed for the timesteps listed in this particular file, overriding the default `ages.dat` (located in the PÉGASE-HR directory `data/user_defined/`). In this case, a warning is given during the execution of `spectra_HR`.

4.4 Measuring Lick indices on high-resolution synthetic spectra

Once high-resolution evolutionary spectra are computed, one can measure the classic Lick indices by running for example

```
> lick myspectrum.fits 5.
```

The first argument is the evolutionary synthetic spectrum; the second argument is the FWHM of the emission lines. In the example above, the predicted emission lines (contained in the extension `ETS_LINES` of the FITS file) are added to the spectrum, with FWHM=5 Å for each line, before measuring the indices. The emission lines are computed as in PÉGASE.2 for a mean HII region, consistently with the ionizing flux of the massive stars.

If one doesn't want to include the emission lines in the measurement, they should set FWHM=0 (by running `lick spectrum 0.`).

If `lick` is called without any command-line argument, then the user is prompted for the filename and the FWHM.

4.5 Measuring colors on low-resolution synthetic spectra

When low resolution synthetic spectra are computed with PÉGASE-HR, one can measure the colors on these spectra, just like in PÉGASE.2. The code to measure the colors is `colors_HR`. It differs from the PÉGASE.2's `color` code only by the input format of the data (FITS format instead of ASCII format).

Run for example:

```
> colors_HR myspectrum.fits
```

to generate an ASCII file containing the evolution of many colors.

`colors_HR` makes use of the files `data/user_defined/filters.dat` and `data/user_defined/calib.dat`. This `calib.dat` file is created automatically by the initial `make` command, with the code `bin/calib_HR` and from the `filters.dat` file (see PÉGASE.2's README for the details of this calibration).

5 References

Fioc M., Rocca-Volmerange B., 1997, A&A 326, 950

Fioc M., Rocca-Volmerange B., 2000, `astro-ph`/9912179 (PÉGASE.2's README)

Katz D., Soubiran C., Cayrel R., Adda, M., Cautain R., 1998, A&A,338,151

Le Borgne D., Rocca-Volmerange B., Prugniel Ph., Lançon A., Fioc M., Soubiran C., 2004, A&A in press, astro-ph/0408xxx

Prugniel Ph., Soubiran C., 2001, A&A,369,1048

Soubiran C., Katz D., Cayrel R., 1998, A&AS,133,221