CIGALE: a python Code Investigating GALaxy Emission⋆

M. Boquien[1], D. Burgarella[2], Y. Roehlly[2], V. Buat[2], L. Ciesla[2], D. Corre[2], A. K. Inoue (井上昭雄)[3], and H. Salas[1]

1 Centro de Astronomía (CITEVA), Universidad de Antofagasta, Avenida Angamos 601, Antofagasta, Chile
  e-mail: `mederic.boquien@uantof.cl`
2 Aix-Marseille Université, CNRS, LAM (Laboratoire d'Astrophysique de Marseille) UMR 7326, 13388 Marseille, France
3 Department of Environmental Science and Technology, Faculty of Design Technology, College of General Education,
  Osaka Sangyo University, 3-1-1 Nakagaito, Daito, Osaka 574-8530, Japan

ABSTRACT

Context. Measuring how the physical properties of galaxies change across cosmic times is essential to understand galaxy formation and evolution. With the advent of numerous ground-based and space-borne instruments launched over the past few decades we now have exquisite multi-wavelength observations of galaxies from the far-ultraviolet (FUV) to the radio domain. To tap into this mine of data and obtain new insight into the formation and evolution of galaxies, it is essential that we are able to extract information from their spectral energy distribution (SED).
Aims. We present a completely new implementation of Code Investigating GALaxy Emission (CIGALE). Written in python, its main aims are to easily and efficiently model the FUV to radio spectrum of galaxies and estimate their physical properties such as star formation rate, attenuation, dust luminosity, stellar mass, and many other physical quantities.
Methods. To compute the spectral models, CIGALE builds composite stellar populations from simple stellar populations combined with highly flexible star formation histories, calculates the emission from gas ionised by massive stars, and attenuates both the stars and the ionised gas with a highly flexible attenuation curve. Based on an energy balance principle, the absorbed energy is then re-emitted by the dust in the mid- and far-infrared domains while thermal and non-thermal components are also included, extending the spectrum far into the radio range. A large grid of models is then fitted to the data and the physical properties are estimated through the analysis of the likelihood distribution.
Results. CIGALE is a versatile and easy-to-use tool that makes full use of the architecture of multi-core computers, building grids of millions of models and analysing samples of thousands of galaxies, both at high speed. Beyond fitting the SEDs of galaxies and parameter estimations, it can also be used as a model-generation tool or serve as a library to build new applications.

Key words. methods: data analysis – methods: numerical – methods: statistical – galaxies: general

1. Introduction

The multi-wavelength emission of galaxies from γ-rays to the radio domain is the outcome of the complex physical interplay between their main baryonic components: stars of all ages and their remnants; molecular, atomic, and ionised gas; dust; and supermassive black holes. This means that the spectral energy distribution (SED) of a galaxy contains the imprint of the baryonic processes that drove its formation and evolution along cosmic times. In other words, to understand galaxy formation and evolution we need to extract the information tightly woven into the SED of galaxies across a broad range of redshifts.

Over the past decade, major efforts have been undertaken to develop and strengthen two of the main pillars upon which rest our studies of galaxy formation and evolution: panchromatic observations and panchromatic models. On the observational side, large multi-wavelength surveys of galaxies have been carried out to measure the SED of galaxies across space and time, yielding a treasure trove of data that provide us with outstanding insight across the different baryonic components of galaxies. In turn, to interpret these observations and measure the fundamental physical properties of galaxies (e.g. star formation rate

(SFR) and history (SFH), stellar mass, attenuation, dust mass and properties, presence and characteristics of an active nucleus, etc.), important investments have been made towards creating ever more precise and accurate models of the emission of galaxies over multiple orders of magnitude in wavelength.

Modelling the SED of galaxies is a heavily intricate problem. Galaxies with very different properties can have broadly similar SEDs. This is particularly the case when considering restricted wavelength ranges rather than the full SED, which is seldomly available. Therefore, estimating the physical properties of galaxies precisely and accurately with only limited data is a considerable challenge. In practice, different avenues can be taken to build physically motivated SED models and attempt to determine their intrinsic physical properties.

A popular approach consists in modelling galaxies using simple dust-attenuated templates representative of the diversity of galaxies at different redshifts. Such an approach is generally adopted by photometric redshift codes that fit only the far-ultraviolet (FUV) to near-infrared (NIR) part of the SED. Although this method is fast and works remarkably well for determining redshifts, as long as spectral breaks are sampled, it shows important limits when it comes to estimating the physical properties of galaxies beyond the stellar mass. In particular it can suffer heavily from degeneracies between the age and the metallicity (e.g. Worthey 1994) or between the age and the

---

⋆ A copy of the code is only available at the CDS via anonymous ftp to `cdsarc.u-strasbg.fr` (`130.79.128.5`) or via `http://cdsarc.u-strasbg.fr/viz-bin/qcat?J/A+A/622/A103`

attenuation (e.g. Papovich et al. 2001): a galaxy can appear red either because it is strongly attenuated, because it does not form stars anymore, or because it has a high metallicity.

A more accurate but much more demanding approach in terms of computational resources is to solve the radiative transfer equation of the emission of stellar populations through a dusty gaseous medium with an arbitrary geometry. While this allows for an exquisitely detailed modelling, the required computing time can be extremely large and the effort required to construct large grids of models rapidly becomes prohibitively expensive even on relatively small samples of galaxies. So far this constraint has largely confined radiative transfer models to theoretical studies (e.g. Gordon et al. 2001; Tuffs et al. 2004; Trayford et al. 2017) and to only a handful of in-depth observational case studies, generally on edge-on galaxies (e.g. Xilouris et al. 1999; Popescu et al. 2000; Bianchi 2008; de Looze et al. 2012), with the modelling of face-on galaxies being a fairly recent development (e.g. De Looze et al. 2014; Viaene et al. 2017).

An increasingly popular compromise in terms of speed, precision, and accuracy is to rely on an energy balance principle: the energy emitted by dust in the mid- and far-IR exactly corresponds to the energy absorbed by dust in the UV-optical range. Such a method has been adopted by modern SED modelling codes such as CIGALE (Burgarella et al. 2005; Noll et al. 2009), MAGPHYS (da Cunha et al. 2008), and FSPS (Conroy et al. 2009; Conroy & Gunn 2010) for instance. Such codes are very versatile and have been applied to study a wide variety of issues: why quiescent galaxies do not follow the starburst IRX–β relation (Boquien et al. 2012), the attenuation properties of galaxies (Buat et al. 2011, 2012, 2018; Boquien et al. 2013; Lo Faro et al. 2017; Salim et al. 2018, Decleir et al., in prep.), SFR estimators (Buat et al. 2014; Boquien et al. 2014, 2016), the separation of the emission of active galactic nuclei (AGNs) from their host galaxy (Ciesla et al. 2015), the imprint of the environment on the SED of galaxies (Bitsakis et al. 2016; Ciesla et al. 2016), or more generally the properties of nearby and distant galaxies (e.g. Burgarella et al. 2011, and in prep.; Giovannoli et al. 2011; Johnston et al. 2015; Małek et al. 2014; Álvarez-Márquez et al. 2016; Pappalardo et al. 2016; Hirashita et al. 2017; Vika et al. 2017; Małek et al. 2018), to cite but a few of the studies carried out with CIGALE. If this approach is so successful, this is largely owing to the efficiency of the method, which allows to obtain good results, breaking the aforementioned degeneracies with the help of dust emission, while doing so rapidly with relatively modest computing requirements.

A key aspect of many modern models is their use of a Bayesian-like approach. The physical properties are then not evaluated from the best-fit model but by weighting all the models depending on their goodness-of-fit, with the best-fit models having the heaviest weight. This naturally takes into account the uncertainties on the observations while also including the effect of intrinsic degeneracies between physical parameters (different models, sometimes with widely different physical parameters, can yield very similar SEDs over some wavelength ranges, making it difficult to favour one model in particular). By doing so we are able to not only convincingly reproduce the observations but also to obtain more reliable estimates of the physical properties and their related uncertainties.

With ever larger and deeper surveys spanning ever broader wavelength ranges, it is especially important that we have equally more efficient, reliable, and versatile tools to model galaxies and estimate their physical properties. We present in this paper the new python version of CIGALE. While it shares the name, the "energy balance" principle, and the Bayesian-like strategy of the original FORTRAN implementation presented in Noll et al. (2009), it is a completely new code that benefits from years of experience developing, maintaining, and using the original CIGALE FORTRAN, while addressing some of the new challenges and usages that have surfaced over the last few years. The aim of this article is to present the new architecture of CIGALE, its different modules, and various examples of its application. For conciseness, we do not dwell on the more theoretical aspects of the Bayesian strategy that have been presented in Noll et al. (2009) and many other articles. Similarly, we do not give excessive details on the precision and accuracy of the results as the topic has already been covered extensively in several papers from the same authors (e.g. Boquien et al. 2012, 2016; Buat et al. 2014; Ciesla et al. 2017). Finally, CIGALE being in constant evolution and development, this article describes its status as of version 2018.1. Further developments will be presented in separate publications.

The article is structured as follows. We present the guiding principles and the architecture of this new code in Sect. 2. The modules used to construct the SED and carry out the analysis are presented in Sects. 3 and 4. The versatility of CIGALE is shown through various examples of its application in Sect. 5. We conclude in Sect. 6. For reference we provide additional technical details, input and output parameters, and an example of a configuration file in the appendices.

2. Architecture

To interpret the results of the modelling of galaxies and avoid a detrimental black-box effect, it is important to understand the whys and wherefores of the model. We present here the broad design principles that we have followed, the reasons for which we have chosen the python language, a high-level overview of the architecture to compute the models and estimate the physical properties of galaxies, and finally some important implementation choices.

2.1. Design principles

While the intrinsic scientific quality of a model is certainly one of the most important criteria determining its impact, other factors also play a role. Ideally, a model should provide clear and meaningful results to a wide population of astronomers from Masters students learning galaxy modelling to highly experienced modellers without requiring a detailed knowledge of the internal mechanics and of the implementation. At the same time a model should remain clear in what it is doing, and how it does it, so that it is flexible and easily adaptable even by inexperienced users, allowing it to easily evolve. Last but not least, the model should not require extraordinary resources. Desktop computers or small departmental servers should be sufficient to analyse large samples of galaxies. To reach these overarching goals, we have designed the python version of CIGALE following three major guiding principles: modularity, clarity, and efficiency both for the users and the developers.

– Modularity: the code must be split into different blocks that are as independent as can be from one another. Each of the four main stages: input handling (e.g. reading and processing the input files), computation of the models (e.g. the fluxes and the physical properties of each model), analysis (e.g. fit of observations and estimation of physical properties), and output handling (e.g. saving the physical properties, the

best-fit spectrum, the χ^2 of each model, etc.), must be entirely independent. Each physical component (stellar populations, nebular emission, attenuation by dust, dust emission, active nucleus, etc.) must be dealt with separately in individual modules, and each module must be able to be substituted as transparently as possible from the point-of-view of upstream and downstream modules. For instance it must be possible to change the attenuation law without affecting the rest of the code in any way. Finally, relying on this modularity, it must be possible to use `CIGALE` not only as initially intended but also as a library to build new tools.

– Clarity: the code must be as easy to understand as can be not only for the developers but also for the users in order to avoid a black-box effect and facilitate the development of community-driven extensions. This is very important to keep the evolution of `CIGALE` in phase with the evolution of knowledge and the creation of new emission models for any physical component existing or newly developed.

– Efficiency: large surveys yield increasingly larger multiwavelength catalogues. We must use computer resources as efficiently as possible in terms of power and memory usage. We aim at being able to model the SED of thousands of galaxies across the universe using millions of models in a matter of a few hours on a typical multi-core computer readily available off-the-shelf.

2.2. Choice of programming language

With these guiding principles in mind, we have chosen to develop the new version of `CIGALE` using the `python` language. We have made this choice based on three main arguments.

1. With its clear syntax and its low barrier of entry, `python` has become an increasingly popular language in Astronomy. It is often the language of choice for teaching programming and has even become the *de facto* standard for many new developments. For `CIGALE`, this means a large fraction of the community is readily able to develop and adapt it to their needs, increasing its potential beyond its original design.

2. A direct cause and consequence of this popularity is that unlike languages more closely tailored for numerical computations such as `FORTRAN` or `idl`, `python` is versatile and has a broad and rich set of specialised and general-purpose libraries. We have relied as much as possible on such libraries, and in particular on `sqlalchemy`[1] for storing models and filters in a database, `numpy` and `scipy` (Oliphant 2007; Jones et al. 2001) for numerical computations, `matplotlib` (Hunter 2007) for plotting, and `astropy` (Astropy Collaboration 2013, 2018) for astronomically related tasks such as computing cosmology-dependent quantities like the luminosity distances or more generally to handle data input and output in a variety of formats, including `FITS` and `VO-table`. This has allowed us to focus our efforts on the scientific challenges rather than on the low level strata of the software.

3. Even though it is a scripting language, the aforementioned scientific modules allow fast numerical computations in `python`, minimising the impact on the performance compared to a compiled language such as C++ or `FORTRAN`. In addition to this, the language comes with a built-in module for parallel programming, allowing for efficient use of multiple cores and processors.

4. Last but not least, `python` is published under a Free license. This means that users do not have to acquire a license to run `CIGALE`, unlike with `idl` for instance. Numerous `python` distributions are available at zero cost with all the required libraries to install and run `CIGALE` easily.

2.3. The division of labour

The primary purposes of `CIGALE` are to generate theoretical models and, optionally, to use them to estimate the physical properties of galaxies. As the latter case is in fact probably the most common situation, we provide here a high-level overview of how this is achieved, only noting the handful of major differences when `CIGALE` is simply used to generate theoretical models.

The `CIGALE` package provides three executable files, each dedicated to a specific task:

1. `pcigale` carries out the computation of the models and if needed the estimation of the physical properties of galaxies.

2. `pcigale-plots` generates plots from the output of `pcigale`: best SED, χ^2 distribution, probability distribution function, and physical properties estimations from mock catalogues.

3. `pcigale-filters` allows to list, delete, add, or plot a filter in the database.

In practice, only `pcigale` is required to create the models, fit them to observations, and estimate the physical properties. The `pcigale-plots` and `pcigale-filters` executables are only provided for convenience and to facilitate the interpretation of the results provided by `pcigale`.

In more detail, `pcigale` handles the optional guided construction of the configuration file (`pcigale init`, which produces a configuration file template where the user then indicates the list of the physical modules to be used, and `pcigale genconf`, which fills the file with the configuration section for each of the user-requested modules), as well as the computation itself (`pcigale run`).

The computation is internally divided into four main stages:

1. Input handling. First the configuration file is read and interpreted: name of the input data files, number of cores, fluxes and properties to fit, parameters for each module, and so on. Then the input data for each object to be analysed are also read: names, redshifts, distances (optional)[2], fluxes, and physical properties (optional). These data are then processed and normalised (e.g. eliminating invalid data, adding missing uncertainties, etc.). If `CIGALE` is only used to generate models, the data file is only used to extract the list of bands.

2. Model generation. For every combination of the input parameters, compute the physical properties of the model (SFR, stellar mass, attenuation, etc.) and its fluxes in a given set of bands.

3. Analysis. For each object: (a) fit all the models to the data, (b) estimate the likelihoods for all the models, and (c) estimate the physical properties from the likelihoods. If only the generation of models is requested, then this step is skipped altogether.

4. Output handling. For each object, save (a) the physical properties estimated from the likelihood distributions, (b) the fluxes and the physical properties of the best-fitting model, and (c), optionally, additional information such as the spec-

---

[1] http://www.sqlalchemy.org/

[2] If the luminosity distance is explicitly provided, it overrides the distance computed from the redshift. The difference can be particularly important for nearby galaxies where peculiar motions dominate over the Hubble flow.

trum of the best model, the χ^2 of all the models, and so on. If only the generation of models is requested, then only the computed fluxes are saved as well as the individual spectra.

After `pcigale run` has completed, it is possible to generate a plot of the best model along with the observations as well as a range of plots related to the evaluation of the parameters with `pcigale-plots`.

3. Model creation modules

The physical processes at play in galaxies provide us with a natural path to build models and compute their physical properties. In `CIGALE`, the models are progressively computed by a series of independent modules called successively, each corresponding to a unique physical component or process. The typical sequence to build each model is the following:

1. Computation of the SFH of the galaxy.
2. Computation of the stellar spectrum from the SFH and single stellar population models.
3. Computation of the nebular emission (lines and continuum) from the Lyman continuum photon production.
4. Computation of the attenuation of the stellar and nebular emission assuming an attenuation law; computation of the luminosity absorbed by the dust.
5. Computation of dust emission in the mid-infrared (mid-IR) and far-IR based an energy balance principle: the energy absorbed by the dust at short wavelengths, which has been computed in the previous step is re-emitted at longer wavelengths.
6. Computation of the emission of an active nucleus.
7. Redshifting of the model and computation of the absorption by the intergalactic medium (IGM).

In practice, the models are progressively computed by successively applying these different modules, each adding a different physical component (spectrum and associated physical parameters). For each model these individual spectral components and the combined spectrum are stored individually to ease the subsequent computation (e.g. to account for the differential reddening between younger and older stellar populations, we need to store these populations separately) and allow the user to easily retrieve the contribution from each physical component. For quantities that are more conveniently computed from the full rest-frame spectrum, in particular those that are directly measured observationally from the spectrum (e.g. line equivalent widths, UV slope β, colours, etc.), a special module can be added prior to redshifting to calculate them on the rest-frame spectrum. We describe here how we have modelled and parametrised each of these different physical components. As new modelling needs appear in the future, we will keep on improving these modules as well as adding new modules whenever necessary, a unique feature derived from the architecture and modularity of `CIGALE`. We should note that in addition to the modules we present here, `CIGALE` also provides unofficial modules that expand its capabilities even further and we support users to develop new modules and encourage them to make them available to the `CIGALE` community.

3.1. Star formation history

As galaxies evolve secularly, accrete and expel gas, or interact with one another over cosmic times, their SFR is expected to vary considerably in non-trivial ways, from episodes of intense star formation to very quiescent phases. Constraining the SFH of galaxies is a tremendously difficult task. Not only because these

variations are so complex, but also because dramatically different SFHs can sometimes yield remarkably similar SEDs. This difficulty to constrain the SFH of galaxies from broadband data has led most studies to adopt relatively simple SFH prescriptions aimed at reproducing the broad variations of the SFR with time: decaying or rising exponentials, delayed, or à la Sandage (1986) for instance. However, with increasingly detailed numerical simulations it is now also possible to adopt more realistic SFH directly derived from such simulations or semi-analytic models (e.g. Pacifici et al. 2012; Boquien et al. 2014). To encompass these two approaches, `CIGALE` handles both analytical SFH depending on several parameters, and arbitrary SFH. In Fig. 1 we present some SFHs obtained from these modules, using a set of parameters representative of their versatility. We note that only one type of SFH (e.g. exponential, delayed, etc.) can be used in a single run. This means that different runs are needed to compare different parametrisations.

3.1.1. Basic assumptions on the SFH

Even though star formation is often modelled as mathematically continuous, it is a fundamentally discrete process, with stars being stochastically born one at a time. Building spectra from the ages of individual stars would rapidly become overwhelming computationally (notwithstanding the fact that we do not know this sort of information beyond the local group), it is therefore reasonable to assume some level of discretisaton on the SFH. In `CIGALE` we introduce two levels of discretisation.

First, we adopt a sampling period of 1 Myr for the SFH. Considering the time t and assuming the age of the galaxy t_0, the sampling grid starts at $t = 0$ Myr and the last sample is at $t = t_0 - 1$ Myr. It is important to note that the SFR is computed at the beginning of an age bin, but the contribution of stars to the spectrum is computed at the end of that age bin.

A sampling of 1 Myr is however too long to capture some brief but important stellar evolutionary phases. We therefore assume that in any given bin, star formation occurs in ten instantaneous episodes separated by 0.1 Myr. For instance if the SFH sampling indicates an SFR of $1\,M_\odot\,\mathrm{yr}^{-1}$ for a given bin, we distribute equally in time ten bursts of $10^5\,M_\odot$. To limit the computation cost of this approach, the single stellar populations presented in Sect. 3.2 are stored with a sampling over an age grid of 1 Myr but already precomputed assuming ten smaller bursts.

Finally, each SFH is automatically normalised so that the total mass of stars formed from the onset of star formation to the last time step is always $1\,M_\odot$. This definition does not correspond to the stellar mass because it does not take into account the return fraction, which depends on the specifics of the stellar populations. We see in Sect. 4.3 how the models are scaled to the observations, which in effect is equivalent to scaling the SFH to the proper level for each observation.

3.1.2. sfh2exp, sfhdelayed, and sfhperiodic modules

We present here three modules defining analytic SFH covering three different general cases: SFH defined by single or double exponentials (`sfh2exp`), delayed SFH with an optional exponential burst (`sfhdelayed`), and periodic SFH (`sfhperiodic`).

sfh2exp. One of the simplest ways to model the SFH of a galaxy is to model it with one or two decaying exponentials. Conceptually, the first exponential models the long-term star formation that has formed the bulk of the stellar mass, whereas the second one models the most recent burst of star formation.
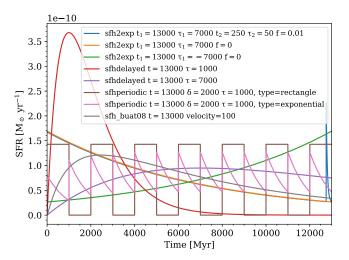
Fig. 1. SFH generated with the `sfh2exp`, `sfhdelayed`, and `sfhperiodic` modules. These represent the cases of two decreasing exponentials (blue), a single decreasing exponential (orange), one increasing exponential (green), a delayed SFH with different timescales (red and purple), a periodic rectangular SFH (brown), a periodic exponential SFH (pink), and the rotation velocity-dependent SFH of Buat et al. (2008; grey). We point out the transitory phase for the periodic exponential as each of the decaying exponentials combine. The exact parameters are indicated in the box. All SFHs have been normalised to have formed 1 $M_\odot$ over 13 Gyr. The diversity of generated SFHs allows for an important flexibility in the modelling.

The combination can be expressed in the following way:

$$\text{SFR}(t) \propto \begin{cases} \exp(-t/\tau_0) & \text{if } t < t_0 - t_1 \\ \exp(-t/\tau_0) + k \times \exp(-t/\tau_1) & \text{if } t \geq t_0 - t_1, \end{cases} \quad (1)$$

with t_1 being the age of the onset of the second episode of star formation relative to t_0 (i.e. if the galaxy started forming stars 13 Gyr ago and had a burst of star formation 100 Myr ago, $t_0 = 13$ Gyr and $t_1 = 100$ Myr), τ_0 and τ_1 the e-folding times of the populations modelling the older stellar populations and the most recent episode of star formation, and k the relative amplitude of the second exponential, which is computed from the burst strength f defined as the fraction of stars formed in the second burst relative to the total mass of stars ever formed. As (a) the SFH is sampled with a period of 1 Myr, (b) we assume a constant SFR between two samples, and (c) by convention we assign the first timestep a time of 0 Myr, f can be expressed in the form of discrete integrals:

$$f = \frac{k \sum_{t=t_0-t_1-1}^{t_0-1} \exp(-t/\tau_1)}{\sum_{t=0}^{t_0-1} \exp(-t/\tau_0) + k \sum_{t=t_0-t_1-1}^{t_0-1} \exp(-t/\tau_1)}, \quad (2)$$

which means that k can be easily computed from the following relation:

$$k = \frac{f}{1-f} \times \frac{\sum_{t=0}^{t_0-1} \exp(-t/\tau_0)}{\sum_{t=t_0-t_1-1}^{t_0-1} \exp(-t/\tau_1)}. \quad (3)$$

Such a formulation, despite its apparent simplicity, is very versatile:
– Very large values of τ compared to t_0 can be used to model a nearly constant SFR.
– Rising exponentials are obtained setting τ to a negative value.

– The classical case of a single exponential can be obtained setting $f = 0$.

This allows for an efficient modelling of elliptical galaxies (case of a single exponential) or of galaxies having had a recent episode of star formation for instance. However, a clear weakness of this module is that it is not adapted for galaxies which have had a recent drop in their SFR, such as galaxies being quenched due to an infall on a cluster.

`sfhdelayed`. The sudden onset of star formation and burst episodes in a double-exponential parametrisation may be too extreme in many practical cases where we expect the variation of the SFH to be smoother. An increasingly popular way to model the SFH of galaxies is the so-called "delayed" SFH:

$$\text{SFR}(t) \propto \frac{t}{\tau^2} \times \exp(-t/\tau) \text{ for } 0 \leq t \leq t_o, \quad (4)$$

with t_o the age of the onset of star formation, and τ the time at which the SFR peaks. Such a functional form has the advantage of providing a nearly linear increase of the SFR from the onset of star formation rather than an abrupt one in the case of `sfh2exp`. After peaking at $t = \tau$, it smoothly decreases.

To allow for more flexibility, the module also allows for an exponential burst representing the latest episode of star formation (Małek et al. 2018). The burst strength is defined following the same concept as for `sfh2exp`, substituting the exponential for the older stellar populations (indices 0) in Eqs. (2) and (3) for the delayed SFH from Eq. (4).

While a delayed SFH allows us to efficiently model early-type (for small τ) and late-type (for large τ) galaxies, one obvious limitation of this functional form is that it does not allow for a recent quenching of the SFR. To address this issue, Ciesla et al. (2017) expanded `sfhdelayed` allowing for an instantaneous recent variation of the SFR, upward or downward, and setting it to a constant until the last time step. This module is provided as `sfhdelayedbq`. This approach was used to successfully model the broad range of KINGFISH galaxies in Hunt et al. (2019).

`sfh_buat08`. Rather than relying a priori on pure analytical functions, an alternative approach has been to tie the SFH to an observed physical quantity of the galaxy. This was done for example by Boissier et al. (2003), Buat et al. (2008) who related the SFH to the rotational velocity of the galaxy. Their SFH is parametrised as:

$$\text{SFR}(t) \propto 10^{a+b\times\log(t)+c\times t^{1/2}}, \quad (5)$$

with t ranging from 1 to t_0 in units of gigayears, and a, b, and c being constants that depend on the rotational velocity of the galaxy. We adopt an extended version of the constants presented in Table 2 of Buat et al. (2008)[3].

`sfhperiodic`. This module provides periodic SFH. The star formation episodes can be of three forms: exponential, delayed, or rectangular. There are four input parameters: 1. the shape of the star formation episodes, 2. δ, the elapsed time between the beginning of each episode of star formation, 3. τ, the duration of each star formation episode, and 4. t_o, the age of the onset of the first star formation episode (i.e. the age of the oldest stars).

3.1.3. `sfhfromfile` module

To build models with arbitrarily complex SFH and combine with hydrodynamical simulations or semi-analytic models, the

---

[3] Samuel Boissier (priv. comm.).

`sfhfromfile` module allows to read and process SFH read from files. The first column of the file contains the age, starting from 0 and with a step of 1 Myr and each subsequent column contains the SFH with the SFR in $M_\odot\,\text{yr}^{-1}$ and a step of 1 Myr for each line. The file can be provided indifferently in `ASCII`, `FITS`, or `VO-table` formats.

To use these SFH, besides the file name, the module requires the indices of the columns to consider, and the ages in millions of years at which the model should be computed. This allows one to compute the SED of a given SFH at different time steps, for instance to investigate its variation with respect to time (Boquien et al. 2014). Optionally, it is also possible to normalise the SFH so that the total stellar mass formed is $1\,M_\odot$ in a similar way as for the `sfh2exp`, `sfhdelayed`, and `sfhperiodic` modules.

3.2. Stellar populations

With the SFH having been computed with one of the modules described in Sect. 3.1, the next step is to compute the intrinsic stellar spectrum. To do so, in addition to the SFH, we need to adopt a library of single stellar populations (SSPs). We rely on two popular libraries of SSPs, that of Bruzual & Charlot (2003; module `bc03`) and that of Maraston (2005; module `m2005`).

Each SSP library is available for a broad range of metallicities (0.0001, 0.0004, 0.004, 0.008, 0.02, and 0.05 for Bruzual & Charlot 2003, and 0.001, 0.01, 0.02, and 0.04 for Maraston 2005) and for two initial mass functions (IMFs; Salpeter 1955 and Chabrier 2003 for Bruzual & Charlot 2003, and Salpeter 1955 and Kroupa 2001 for Maraston 2005).

The Bruzual & Charlot (2003) SSPs come in low and high resolution versions, both of which are provided with `CIGALE`. By default the low-resolution models are used as they are generally sufficient for use with broadband data. An option is provided to build the database with high-resolution models, which are useful for instance when dealing with narrow features such as absorption or emission lines.

To compute the spectrum of the composite stellar populations, we calculate the dot product of the SFH with the grid containing the evolution of the spectrum of an SSP with steps of 1 Myr[4]. In Fig. 2 we show the results of this computation using the `bc03` and `m2005` modules.

At this stage the stellar populations are dust-free. We need however to anticipate that there can be a differential reddening between young stellar populations embedded in their dust clouds and older populations that have escaped and are therefore less reddened (e.g. Charlot & Fall 2000). To account for this, we compute and store separately the spectra of old and young stars so they can be attenuated independently in a downstream module (Sect. 3.4). Following the prescription of Charlot & Fall (2000), the default age of separation between these populations is 10 Myr but it can be configured freely.

3.3. Nebular emission

The most massive stars emit a significant fraction of their light in the Lyman continuum. These high-energy photons ionise the surrounding gas which re-emits the energy in the form of a series of emission lines and a continuum (free-free, free-bound, and two-photon) that extends far into the radio regime. This emission is important as it provides us with excellent probes into the most recent star formation through hydrogen lines
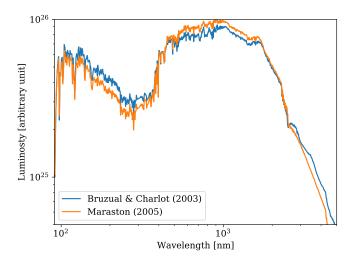
---

[4] Taking SFH(t) to be the SFR at age t, and SSP(λ, t) to be the spectrum of a single population at wavelength λ and of age t, as the age grid is regular and identical, the object spectrum S(λ) is $\sum_t \text{SSP}(\lambda, t) \times \text{SFH}(t)$.

Fig. 2. Spectra of the composite stellar populations for the Bruzual & Charlot (2003; blue) and Maraston (2005; orange models). Both models have an identical SFH generated with the `sfh2exp` module ($t_0 = 13\,000\,\text{Myr}$, $\tau_0 = 7000\,\text{Myr}$, $t_1 = 250\,\text{Myr}$, $\tau_1 = 50\,\text{Myr}$, $f = 0.01$), a Salpeter (1955) IMF, and a metallicity $Z = 0.02$. Even if the Maraston (2005) models have been developed to handle the contribution of thermally pulsating asymptotic giant branch stars, there are also clear differences at all wavelengths. Such differences are strongly dependent on the actual SFH and the differences seen in this plot are not necessarily representative of what would be obtained with another SFH.

and radio continuum as well as the gas metallicity from metal lines. While the nebular emission generally contributes little to the broadband fluxes of quiescent star-forming galaxies, and is therefore ignored, this is not the case at the local scale when considering starbursting dwarf and very young star-forming regions (e.g. Anders & Fritze-v. Alvensleben 2003; Boquien et al. 2010) as well as some high-redshift galaxies (e.g. Stark et al. 2013; de Barros et al. 2014). This has a direct impact on SED modelling and the nebular emission needs to be carefully taken into account.

To model the nebular emission in `CIGALE` we have adopted nebular templates based on Inoue (2011), which have been generated using `CLOUDY 13.01` (Ferland et al. 1998, 2013). They predict the relative intensities of 124 lines from H II regions from He II at 30.38 nm to [N II] at 205.4 μm. These templates are parametrised according to ionisation parameter U, and the metallicity Z, which is assumed to be the same as the stellar one. The electron density is assumed to be constant and is set to $100\,\text{cm}^{-3}$. Important improvements compared to the original templates of Inoue (2011) include a refinement of the sampling in $\log U$ to steps of 0.1 dex, the extension down to $\log U = -4$, and changes in the abundances. The abundance set is based on the Orion nebula. The helium and nitrogen abundances are scaled by metallicity following Nagao et al. (2011). This is motivated by the fact that helium has a primordial abundance floor and the nitrogen production is dominated by the secondary nucleosynthesis process through the CNO cycle at high metallicity.

In practice the computation of the nebular emission in `CIGALE` follows several steps. First of all, after having selected a given template (based on U, and Z), which gives line luminosities normalised to the ionizing photon luminosity, the spectrum of emission lines is computed. Each line has a Gaussian shape with a user-defined line width to take gas motion into account, which can be especially important for narrow-band filters and

high-redshift objects due to the apparent line broadening with redshift in the observed frame. While this gives the normalised nebular emission line spectrum which could be rescaled to the appropriate level by multiplying by the ionizing photon luminosity which was computed along with the composite stellar population, this would ignore the fact that not all Lyman-continuum photons ionize the surrounding gas. Two main processes can affect the ionisation rate of the surrounding gas. First of all, a fraction of the Lyman continuum can simply escape from the galaxy. Even though the escape fraction is generally low in the nearby universe, it may reach much higher values at high redshift to reionise the universe (e.g. Inoue et al. 2006; Hayes et al. 2011). The other process that can prevent Lyman continuum photons from ionising the surrounding gas is absorption by dust (Inoue 2001; Inoue et al. 2001). In this case it contributes to the general dust heating and is accounted for in the dust emission models presented in Sect. 3.5. To take these two processes into account, we downscale the nebular line spectrum by the following factor from Inoue (2011):

$$k = \frac{1 - f_{\rm esc} - f_{\rm dust}}{1 + \alpha_1 (T_{\rm e}) / \alpha_B (T_{\rm e}) \times (f_{\rm esc} + f_{\rm dust})}, \qquad (6)$$

with α_B being the case B recombination rate in $m^3\,{\rm s}^{-1}$, $\alpha_1 = \alpha_A - \alpha_B$, the recombination rate to the ground level, $T_{\rm e}$ the electron temperature in K, $f_{\rm esc}$ the Lyman continuum escape fraction and $f_{\rm dust}$ the partial absorption by dust before ionisation. Numerically, for $T_{\rm e} = 10^4$ K, we take $\alpha_B = 2.58 \times 10^{-19}$ m^3 s^{-1} $\alpha_1 = 1.54 \times 10^{-19}$ m^3 s^{-1} (Ferland 1980).

The nebular continuum is computed following the prescription by Inoue (2010) with the same parameters as the emission line templates and is computed in a similar fashion, including normalisation to the Lyman continuum photon luminosity and correction for the escape fraction and absorption by dust. Only the hydrogen continuum is taken into account as helium and other metal element continua are weak and negligible.

It is important to note that the nebular emission does not consider the emission of metal and CO lines in photo-dissociation regions and molecular clouds. In effect, [C ɪɪ] at $158\,\mu$m or [O ɪ] $63\,\mu$m/$145\,\mu$m are seriously underestimated for galaxies with photo-dissociation regions. This will be considered for a future version of the code.

We present a model of an SED including nebular emission in Fig. 3.

3.4. Attenuation laws

Galaxies contain dust, and this dust is very efficient at absorbing short-wavelength radiation. The energy absorbed from the UV to the NIR is then re-emitted in the mid- and far-IR. This energy balance principle lies at the core of `CIGALE`. It is therefore important that the attenuation is properly modelled.

First, we have to distinguish between an extinction curve, which is only dependent on the dust grain mix (composition, size distribution, etc.), and an attenuation curve, which also depends on the geometry. Except for a handful of very nearby objects such as the Magellanic Clouds, observers only see the effect of attenuation laws in galaxies. Because attenuation laws change with redshift and from galaxy to galaxy, `CIGALE` needs to be able to cover a broad range of such laws both in terms of shape and in terms of normalisation.

The most direct approach would be to consider a mix of dust grains and a geometry. However as we see below, we use different sets of templates to model dust emission. Each template

would therefore need to have a specific extinction curve corresponding to the assumed mix. This would be difficult for empirical templates that do not assume specific grain properties. The assumption of a geometry is also a problem as `CIGALE` can be used on vastly different objects, from small regions in galaxies (down to sub-kpc scale) to large galaxies at all redshifts. In any case, observations of the Milky Way show that the relative distribution of dust and stars can be much more complex than the simple geometries that are often assumed (slab, sandwich, shell, etc.) and would therefore require a full radiative transfer with a realistic geometry (e.g. De Looze et al. 2014).

An indirect but suitably much faster approach is to assume attenuation laws. Numerous studies have focussed on determining attenuation laws in galaxies, finding a remarkable diversity (e.g. Wild et al. 2011; Reddy et al. 2015, 2016; Lo Faro et al. 2017; Buat et al. 2018; Salim et al. 2018, Decleir et al., in prep., and many others). This means that these attenuation laws must be flexible so that they can adapt to the broad diversity of observed curves. In this endeavor, we have pursued two ways of modelling attenuation curves in galaxies: the implementation of the Charlot & Fall (2000) model, and flexible laws inspired from the starburst curve (Calzetti et al. 2000).

3.4.1. The `dustatt_modified_CF00` module

As a first approach to addressing this problem, we implemented the Charlot & Fall (2000) model through the `dustatt_modified_CF00` module. The key idea behind this model is the realisation that not only young stars still embedded in their birth cloud suffer from additional attenuation compared to stars that have broken out and escaped into the ISM, but also that the attenuation curves associated to the birth cloud and the ISM must be different. In practice, this is modelled by assuming two different power-law attenuation curves of the form $A(\lambda) \propto \lambda^\delta$: one for the birth cloud with a default slope of $\delta_{\rm BC} = -1.3$, and one for the ISM with a default slope of $\delta_{\rm ISM} = -0.7$. Because radiation from young stars has to travel through both the birth cloud and the ISM to escape the galaxy, the spectrum of stars younger than 10 Myr are attenuated by both the birth cloud and ISM curves. Stars older than 10 Myr are only attenuated by the ISM curve. Following Sect. 3.2, this age can be configured freely through the stellar populations module. In each case the nebular emission is attenuated following the same law as the stars giving rise to it. The V-band attenuation of both curves are linked through the relation:
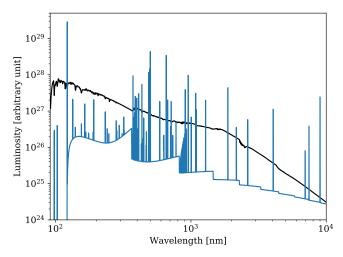
$$\mu = \frac{A_V^{\rm ISM}}{A_V^{\rm BC} + A_V^{\rm ISM}}, \qquad (7)$$

or, in other words, the ratio of the total attenuation undergone by stars older than 10 Myr to that undergone by stars younger than 10 Myr. This module is flexible beyond a strict implementation of the Charlot & Fall (2000) model in the sense that $A_V^{\rm ISM}$, μ, δ_{BC}, and $\delta_{\rm ISM}$ are all input parameters.

It should be noted that `CIGALE` also provides the module `dustatt_powerlaw`, which should not be confused with this module as it departs in several ways from the Charlot & Fall (2000) model, having a single power law for both young and old stars, only with a different absolute attenuation, and the attenuation is set as the total attenuation for each component.

3.4.2. The `dustatt_modified_starburst` module

A more empirical approach is to use a well-known curve as a baseline. Subsequently, this curve can be parametrised to
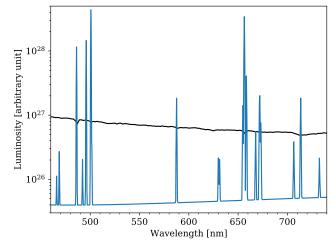
Fig. 3. *Left panel*: nebular (blue) and stellar (black) FUV to NIR spectra. In total 124 lines from H II regions are taken into account. The nebular continuum takes into account free-free, free-bound, and two-photon processes. The lines are modelled from an improved version of the CLOUDY templates computed in Inoue (2011). Here the electron density is set to 100 cm^{-3}, the metallicity $Z = 0.02$, and the ionisation parameter $\log U = -2$. Both f_{esc} and f_{dust} are set to 0. The stellar emission is computed with the sfh2exp ($t_1 = 13000$ Myr, $\tau_1 = 7000$ Myr, $t_2 = 25$ Myr, $\tau_2 = 50$ Myr, and $f = 0.1$) and bc03 (Salpeter 1955; IMF and $Z = 0.02$) modules. *Right panel*: zoom in the 460 nm–740 nm range. At this scale we can see the line width due to gas motions. Here the line width is set to 300 km s^{-1}.

make it more generic and allow for some flexibility, for example in terms of slope or to account for the presence of a bump around 220 nm. We have also adopted this solution with the dustatt_modified_staburst module. It is based on the Calzetti et al. (2000) starburst attenuation curve, extended with the Leitherer et al. (2002) curve between the Lyman break and 150 nm. Its slope can be modified by multiplying it by a power law function of slope δ similar to the one described above and a UV bump can be added. This bump is modelled as a Lorentzian-like Drude profile which is described by three parameters: its central wavelength, its full width at half maximum (FWHM), and its amplitude (see Eq. (3) from Noll et al. 2009). The overall attenuation can be expressed as:

$$k_\lambda = D_\lambda + k_\lambda^{\mathrm{starburst}} \times (\lambda/550\,\mathrm{nm})^\delta \times \frac{E(B-V)_{\delta=0}}{E(B-V)_\delta}, \qquad (8)$$

with D_λ the Drude profile, and the last term renormalising the curve so that $E(B-V)$ remains equal to the input $E(B-V)$ when $\delta \neq 0$. We show a set of stellar attenuation curves representative of the flexibility of our approach in Fig. 4.

Formally, the starburst law is defined for the continuum only, the emission lines being dimmed with a Milky Way extinction law. Here we have adopted a slightly more flexible approach, adopting the Milky Way curve of Cardelli et al. (1989) with the O'Donnell (1994) update, as well as the Small and Large Magellanic Cloud extinction curves of Pei (1992). The value of R_V can be modified. The overall normalisation of the curves affecting the stars and the lines is determined according to the reddening of the emission lines $E(B-V)_{\mathrm{lines}}$, with a simple reduction factor f between the two curves:

$$f = \frac{E(B-V)_{\mathrm{continuum}}}{E(B-V)_{\mathrm{lines}}}, \qquad (9)$$

with lines being more dimmed by dust than stars.

3.5. Dust emission

As dust absorbs stellar photons from the UV to the NIR, this energy is re-emitted at longer wavelengths, essentially in the
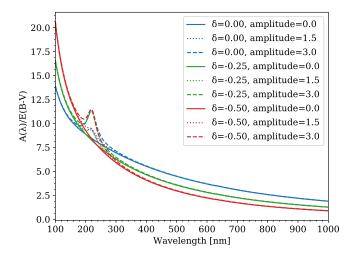

Fig. 4. Three stellar attenuation curves generated by the dustatt_modified_starburst module of CIGALE. Based on the Calzetti et al. (2000) attenuation law, the red, green, and blue lines correspond to a power-law modification with indices δ of 0.00, -0.25, and -0.50. In addition, a 220 nm bump has been added with three different amplitudes: 0 (dotted), 1.5 (solid), and 3 (dashed). The difference in the normalisation comes from the fact that $E(B-V)$ is kept constant after multiplying the starburst law (case $\delta = 0$) by a power law.

mid- and far-IR domains. In general, dust emission can be split into three broad components. In the mid-IR, around 8 μm, the emission is dominated by polycyclic aromatic hydrocarbon (PAH) bands. At longer wavelengths, the emission is progressively taken over by very small, warm grains that tend to be stochastically heated for weak and moderate radiation field intensities but progressively becomes dominated by equilibrium emission at higher intensities. Beyond $\sim$100 μm, the emission is increasingly due to big, relatively cold grains. The different heating mechanisms of these different dust species, their composition, the metallicity, and the intensity and shape of the incident radiation field, all have an impact on the dust SED.

The modelling of dust emission is a very active domain of research, building on several generations of increasingly more powerful IR instruments, from IRAS to *Herschel*. For `CIGALE`, we consider three different sets of models: the Dale et al. (2014) empirical templates, the Draine & Li (2007) models (including the updates of Draine et al. 2014), and the Casey (2012) analytic model. The modules are described hereafter and some examples of dust SED are shown in Fig. 5.

3.5.1. `dale2014` module

The dust templates of Dale et al. (2014) are based on a sample of nearby star-forming galaxies originally presented in Dale & Helou (2002). The latest update refines the PAH emission and also adds an optional AGN component as seen in Sect. 3.7. Aiming at simplicity, the star-forming component is parametrised by a single parameter α defined as: $dM_d(U) \propto U^{-\alpha}dU$, with M_d being the dust mass, and U the radiation field intensity. The α parameter is itself tightly linked to the 60–100 μm colour. The main strength of this model is its simplicity, with only one easy-to-interpret parameter. However, the PAH emission relative to the total infrared shows a limited variation with respect to α. This can be an issue in particular in metal–poor galaxies which are known to have only little PAH emission (e.g. Engelbracht et al. 2005).

3.5.2. `dl2007`, `dl2014`, and `themis` modules

Presented in Draine & Li (2007), these models are based on a dust mixture of amorphous silicate and graphite grains, and PAH. One of the key features of the Draine & Li (2007) templates is the separation of dust emission into two components. The first one models the diffuse dust emission heated by the general stellar population. In this context, the dust is illuminated with a single radiation field U_{min}. The second one models dust tightly linked to star-forming regions. In that case the dust is illuminated with a variable radiation field ranging from U_{min} to U_{max} following a power-law index α (see Eq. (23) of Draine & Li 2007). By default it is set to a fixed value of $\alpha = -2$. The dust mass fraction of dust linked to the star-forming regions (respectively diffuse emission) is γ (respectively $1 - \gamma$). The last parameter of these models is q_{PAH}, the mass fraction of the PAH, which is common for the two components. These components are kept separate in `CIGALE` to allow for their individual inspection.

In recent years, this model has been refined further. Because the parameters are not identical, a different module is available to use the updated models: `dl2014`. Among the main differences to note, these new models have led to the following: 1) an expansion on the range of radiation field intensities and PAH mass fractions; 2) the power law index α is now a free parameter; 3) U_{max} has been set to 10^7; 4) a change in the treatment of graphite; 5) the dust masses have been renormalised (Draine et al. 2014).

Following the same parametrization, the DustPedia team has developed a module using the THEMIS dust model (Jones et al. 2017). The module and its application are described in detail in Nersesian et al. (in prep.).

One of the main strengths of these models is that they are very flexible. They can account for very different physical conditions with a variety of radiation fields and a variable PAH emission. However, this flexibility comes at the cost of a much larger parameter space to explore compared to the Dale et al. (2014) templates and is therefore more expensive in terms of processing power and memory.

3.5.3. `casey2012` module

The `casey2012` module implements the analytic model of Casey (2012). Dust emission is modelled with two components: a single temperature modified black body in the FIR "representing the reprocessed starburst emission in the whole galaxy" and a power law in the mid-IR "which approximates hot-dust emission from AGN heating or clumpy, hot starbursting regions" (Casey 2012). In practice, the module depends on three parameters: the temperature of the dust, the emissivity index of the dust, and the mid-IR power law index. To distinguish both components and easily assess their relative contributions, `CIGALE` stores them separately in the SED.

While less physically motivated than the Draine & Li (2007) models and not based on observations as the Dale et al. (2014) templates, the Casey (2012) models are very flexible and can be easily used for local and high-redshift galaxies. The main limitations of this module, however, are that it includes no PAH emission and that the IR is computed from an energy balance and thus AGN heating is in effect not included.

3.6. Synchrotron radio emission

With the advent of the Square Kilometre Array, an avalanche of data in the centimetre regime is upon us. At such wavelengths, the emission is split between thermal processes related to the ionisation of the gas by massive stars and non-thermal processes related to the interaction of relativistic electrons from supernovae with the local magnetic field.

While the `nebular` module naturally models the thermal radio continuum, it lacks synchrotron emission. The exact shape and intensity of the synchrotron spectrum depends on various parameters such as the strength of the magnetic field, the energy of the relativistic electrons propagating through it, and so on. Rather than attempting to model the synchrotron in such detail, the `synchrotron` module relies on the radio-IR correlation q_{IR} of Helou et al. (1985), a free power-law spectral slope α, and on the assumption that at 21 cm the spectrum is largely dominated by non-thermal emission. In effect, knowing the IR emission, q_{IR} directly provides the luminosity density at 21 cm. It is then a simple matter of computing a spectrum with the requested α and scaling it so that it matches the estimated luminosity. On the other hand, radio data can help to estimate the IR emission if no other data are available in this range.

3.7. Active galactic nuclei

Along with star formation, AGNs are thought to have a dramatic impact on galaxy evolution. Yet, properly disentangling the emission of AGNs from star formation is not necessarily an easy task as they can both strongly emit in the UV, and a large fraction of this emission can be reprocessed by dust and re-emitted at longer wavelengths.

Several options are available in `CIGALE` to model the presence of an AGN from coarse but rapid methods to more detailed but slower methods. If only the IR is fitted, the `casey2012` module can be used, with the AGN being simply parametrised by the slope of the power law α. If the AGN is a quasar, the `dale2014` module provides a simple template from the UV to the IR. The AGN is simply parametrised with the AGN fraction defined as the ratio of the AGN luminosity to the sum of the AGN and dust luminosities. While these methods are rapid and easy to use, they do not necessarily offer all the flexibility one may want to take into account the variety of AGN SED. `CIGALE` also
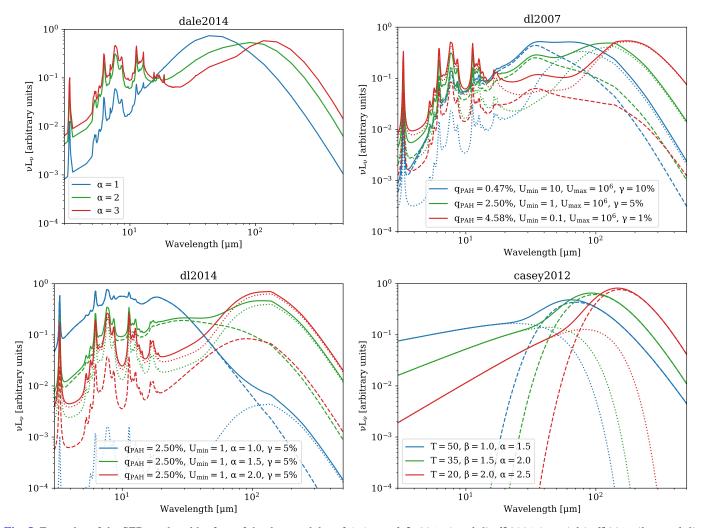
Fig. 5. Examples of the SED produced by four of the dust modules of `CIGALE`: `dale2014` (*top-left*), `dl2007` (*top-right*), `dl2014` (*bottom-left*), and `casey2012` (*bottom-right*). Each colour indicates a different set of parameters shown in the bottom-right corner. The solid lines represent the total SED, summing up all components. For the `dl2007` and `dl2014` modules, the dashed line corresponds to the star-forming regions and the dotted line to the diffuse emission. Finally, for the `casey2012` module, the dashed line corresponds to the modified black body whereas the dotted line corresponds to the power law.

provides the detailed AGN models of Fritz et al. (2006). It explicitly takes into account three components through a radiative transfer model: the primary source located in the torus, the scattered emission by dust, and the thermal dust emission. These modules are determined through a set of seven parameters: r the ratio of the maximum to minimum radii of the dust torus, τ the optical depth at $9.7\,\mu$m, β and γ describing the dust density distribution ($\propto r^{\beta}e^{-\gamma|\cos\theta|}$) with r the radius and θ the opening angle of the dust torus, ψ the angle between the AGN axis and the line of sight, and the AGN fraction. We show some examples of the SED generated by the `fritz2006` module in Fig. 6.

3.8. Measuring physical properties from the rest-frame spectrum

The previous modules have allowed us to build a full FUV-to-radio rest-frame spectrum. If we have the physical properties (SFR, attenuation, stellar and dust mass, etc.) associated with each of the components, some observed quantities can only be accurately computed from the full spectrum. The aim of this module is to compute such quantities before redshifting the spectrum. The following quantities are measured:

- The UV slope β is computed by fitting a straight line to the F_λ spectrum in log–log space over the wavelength ranges defined in Table 2 of Calzetti et al. (1994).
- The D_n4000 break index is computed from the ratio of the mean fluxes of the F_λ spectrum from the rest-frame 400–410 nm on the red side and from 385 to 395 nm on the blue side (Balogh et al. 1999).
- IRX is computed as the log of the ratio of the dust to rest-frame far-UV (GALEX band) luminosities.
- Rest-frame equivalent widths are computed as the ratio of the integral of the spectrum over a user-defined wavelength range, including and excluding nebular lines.
- Rest-frame luminosity densities and colours are computed integrating the spectrum over any filter or pair of filters that are present in the filters database.

3.9. Redshifting

Finally, the last module called to build the SED is the `redshifting` module. It has two effects. First, it redshifts the spectrum and dims it, multiplying the wavelengths by $1 + z$ and dividing the spectrum by $1 + z$. The second effect of this module is to take into account the absorption of short wavelength radi-
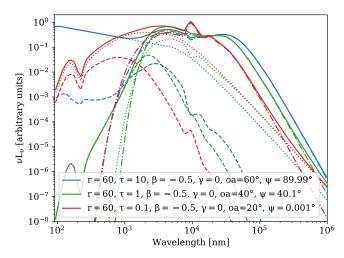
Fig. 6. Examples of the SED produced with the `fritz2006` module. Each colour indicates a different set of parameters shown in the bottom-left corner. The solid lines represent the total emission. The dotted lines represent the AGN accretion disk, the dashed lines the scattered component, and the dash-dot line the thermal emission.
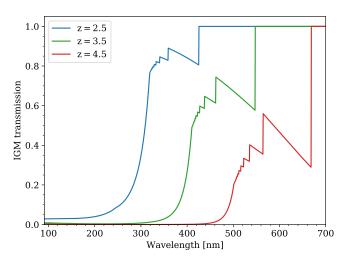

Fig. 7. IGM transmission prescription of Meiksin (2006) in the observed frame. The three redshifts are indicated in the top-left corner.

ation by the IGM. To do so `CIGALE` applies the prescription of Meiksin (2006), which is shown in Fig. 7.

We note that this module takes only one parameter, the redshift. However if the redshifts to apply are not indicated in the configuration file, `CIGALE` then automatically builds a list of redshifts from the input flux file, rounding them to two decimal places by default in order to avoid computing models with many close redshifts. Nevertheless, the physical quantities are computed for the exact input redshift at full precision.

4. Analysis modules

The modules presented in Sect. 3 are the building blocks to compute individual models. But these building blocks alone do not provide us with the desired set of SEDs nor do they provide us with estimates of the physical properties of the objects under consideration. Such tasks lie upon the so-called analysis modules. Two analysis modules are available with `CIGALE`: `savefluxes` to generate a grid of models and save the outputs (fluxes, SFH, physical properties, etc.) and

`pdf_analysis` that not only generates a grid of models but also fits these models to observations to estimate various physical properties and save the outputs.

4.1. Computing physical quantities

The computation of the physical quantities in `CIGALE` depends on their nature. Intrinsic intensive and extensive physical properties are computed in the different physical modules presented in Sect. 3 where it makes most sense. Fluxes however depend on the observer and are measured after the computation of a given model. The technique differs whether we compute fluxes in bandpasses or whether they are extracted from low- or high-resolution spectra.

4.1.1. Bandpasses

The fluxes in bands are computed by integrating the model spectrum through the corresponding filter. The basic method is standard:

$$f_\lambda = \frac{\int_{\lambda_{\mathrm{low}}}^{\lambda_{\mathrm{high}}} F_\lambda(\lambda) \times T(\lambda)\, \mathrm{d}\lambda}{\int_{\lambda_{\mathrm{low}}}^{\lambda_{\mathrm{high}}} T(\lambda)\, \mathrm{d}\lambda}, \qquad (10)$$

with f_λ being the flux per wavelength through a filter of transmission T in units of energy[5] defined between wavelengths λ_{low} and λ_{high}, and F_λ the flux per wavelength of the model spectrum. To preserve the best sampling and not miss features, both the spectrum and the transmission filter are interpolated on one another's wavelengths.

We have to note that the denominator does not depend on the model spectrum and is a constant. To avoid its computation, the transmission curves are normalised in such a way that $\int_{\lambda_{\mathrm{low}}}^{\lambda_{\mathrm{high}}} T(\lambda)\, \mathrm{d}\lambda = 1$.

The observed fluxes are however provided in units of frequency. The flux per wavelength from the integral can easily be converted to fluxes per frequency through the use of the pivot wavelength λ_{pivot} that is independent of the source (Koornneef et al. 1986):

$$f_\nu = \frac{\lambda_{\mathrm{pivot}}^2}{c} f_\lambda, \qquad (11)$$

with c the speed of light in the vacuum and λ_{pivot} defined as:

$$\lambda_{\mathrm{pivot}} = \sqrt{\frac{\int_{\lambda_{\mathrm{low}}}^{\lambda_{\mathrm{high}}} T\, \mathrm{d}\lambda}{\int_{\lambda_{\mathrm{low}}}^{\lambda_{\mathrm{high}}} T/\lambda^2\, \mathrm{d}\lambda}}. \qquad (12)$$

In effect, as $\lambda_{\mathrm{pivot}}/c$ is a constant, we also integrate it to the normalisation of T and we finally rescale the filter so that the integration of a spectrum in flux per wavelength provides a flux density in mJy.

4.1.2. Emission lines

Photometric observations in bandpasses are fairly straightforward to deal with as can be seen above. Measuring emission lines is however more difficult. This is due in no small part to the fact that while bandpasses encompass all of the emission within

---

[5] If the filter is provided in units of photons, it is converted and stored in units of energy in the database of `CIGALE`.

their range, the continuum has to be subtracted from emission lines. The presence of sometimes strong absorption lines under emission lines makes this a difficult problem and the spectral resolution of the data leads to different strategies.

Low resolution data. Low resolution spectra and narrow-band observations do not allow to measure the underlying absorption lines. A commonly used technique is to measure the level of the continuum around the emission line and therefore only take into account the flux above the inferred continuum under the line. This has a natural downside of not taking into account the loss of flux due to the underlying absorption line. To model this we have implemented special filters that naturally subtract the continuum. Each filter is made of a positive part with the transmission set to 1 on the emission part. Off the line it is set to a negative value such that $\int_{\lambda_{\text{low}}}^{\lambda_{\text{high}}} T(\lambda) \, d\lambda = 0$. The flux f is then directly obtained through $f = \int_{\lambda_{\text{low}}}^{\lambda_{\text{high}}} F_\lambda(\lambda) \, T(\lambda) \, d\lambda$ without further normalisation of the filter as we compute a flux rather than a flux density. In effect, the integration of the spectrum on the negative part of the filter will evaluate the flux provided by the continuum and will allow us to subtract it from the flux computed by integrating the positive value. Assuming that the continuum is evaluated well enough, the remainder will be the flux from the line.

For more flexibility, `CIGALE` provides these filters at different spectral resolutions for the main rest-frame optical lines. To compute the line fluxes from the spectra at any redshift the filters are stretched in wavelength by a factor $1 + z$. This stretching is necessary to ensure that each filter remains centred on the line with the same resolution as in the rest frame.

High-resolution data. If the data are at high resolution or if the fluxes have been corrected for absorption lines, the previous technique would not provide reliable results. In such a case `CIGALE` computes the emission line fluxes directly from their theoretical emission based on the nebular emission templates presented above. After normalisation to the number of Lyman continuum photons and extinction by dust, they provide the luminosity in any of the following lines that can be listed in the `pcigale.ini` file: Ly-alpha, CII-133.5, SiIV-139.7, CIV-154.9, HeII-164.0, OIII-166.5, CIII-190.9, CII-232.6, MgII-279.8, OII-372.7, H-10, H-9, NeIII-386.9, HeI-388.9, H-epsilon, SII-407.0, H-delta, H-gamma, H-beta, OIII-495.9, OIII-500.7, OI-630.0, NII-654.8, H-alpha, NII-658.4, SII-671.6, and SII-673.1.

4.2. *savefluxes module*

Our ability to compare observations to theoretical models is key to constraining models of galaxy evolution. This can take various forms, from simple colour–colour plots to the computation of the SED of galaxies in numerical simulations and semi-analytic models. The `savefluxes` module has been designed for this kind of application: it aims at computing and saving the spectra and the properties of arbitrary theoretical galaxies. In practice the steps taken are the following.

 1. From the list of parameters of each SED creation module given in the configuration file (see e.g. Appendix E), `savefluxes` determines the complete list of parameters for each model to be computed. This essentially consists in finding all the possible combinations of parameters, creating the equivalent of an n-dimensional grid, with each dimension corresponding to an individual parameter. Alternatively, the

parameters for each model can be explicitly provided in a file (one line per SED and one column per parameter). The former approach is useful to compute a systematic grid of theoretical models whereas the latter is more adapted for galaxies from simulations whose properties do not follow a grid.
 2. For each model, the spectrum is computed, and its physical properties (both input and derived) and fluxes in passbands (which can be narrow as well as broad) are stored in memory. Optionally, the full spectrum along with the individual components (stellar populations, nebular emission, dust emission, etc.) and the SFH are saved to disk as FITS tables.
 3. The integrated fluxes and the physical properties of all the models are saved to disk both as `ASCII` and `FITS` tables.

4.3. *pdf_analysis module*

The SEDs of galaxies contain a treasure trove of information on their physical properties, which we need to access to understand how they form and evolve. To do so, the simplest and probably most common method consists in fitting the observed SED of a galaxy with a set of models and selecting the best-fitting one. Unfortunately, this method suffers from severe drawbacks. First of all, it ignores the degeneracies one can encounter. Models with almost equally good fits can have very different properties. As such the properties corresponding to the best fit are not necessarily representative of the true properties of the object. A related issue revolves around the estimation of the uncertainties on the physical properties. The best fit in itself does not provide information on the uncertainties. Methods such as bootstrap can be applied at the cost of repeating the fitting procedure numerous times.

A technique that has become increasingly popular over the past decade to address these issues is to rely on the goodness of fit of all the models rather than just the best-fitting model. This is generally done through the likelihood. In this case each model in the grid of models (the priors) will have an associated likelihood taken as $\exp\left(-\chi^2/2\right)$. These likelihoods can then be used as weights to estimate both the physical parameters (the likelihood-weighted means of the physical parameters) and the related uncertainties (the likelihood-weighted standard deviations of the physical parameters). This method works well when the probability distribution function is well behaved (e.g. a single peak). For more difficult cases, the marginalised probability distribution function (PDF) provides the full information to estimate the physical properties.

With either method, an important difference between various fitting codes is the algorithm to sample the priors. Two main strategies are commonly considered. Some codes use a Monte-Carlo Markov Chain (MCMC) method (or some variant of it). This is especially the case when the dimension of the problem is very large, for instance when considering non-parametric SFH (the SFH does not follow any given functional form but rather the SFR is free at every single time step) to fit spectra. While the evident upside is that it allows a large volume of priors to be explored, the sampling can be sparse, with the risk of missing some high-likelihood regions. In addition, the SEDs have to be recomputed (or at least reinterpolated on a grid of precomputed priors) at each step of the exploration of the parameter space and this for each object, which may require particularly long computing times for large catalogues.

An alternative method that we adopt in `CIGALE` is to rely on a fixed grid of models. The main advantage is that the models need to be computed only once for all the objects. Because of

this, numerous optimisations can be applied to compute the grid of models and to fit them to the data. The main downside is that it can be somewhat memory intensive. To get good results, the grid of models needs to be reasonably well sampled. At the same time, for the process to be computationally efficient, the grid along with the associated physical properties to estimate need to remain in memory. The amount of memory that is required primarily depends on 1) the number of models to compute, 2) the number of bands and physical properties to fit, and 3) the number of physical properties to estimate.

In the `pdf_analysis` module we implemented the estimation of the physical properties from likelihood-weighted parameters on a fixed grid of models. The computation of the grid of SEDs and associated physical properties follows the same steps as for the `savefluxes` module described in Sect. 4.2, except for minor differences of no consequence here. Once the grid of models has been computed, the high level algorithm to estimate the physical properties is the following.

1. From the complete set S_0 of models, selection of the subset S_1 of models closest to the rounded redshift of the analysed object. By default, the rounding is to two decimal places, but this can be user-defined.
2. Computation of the multiplicative factors (Eq. (13)) to scale the S_1 models to the observations.
3. Computation of the χ^2 (Eq. (14)) between all the S_1 models and the observations.
4. Computation of the likelihood $\exp\left(-\chi^2/2\right)$ for the S_1 models.
5. Estimation of each physical property along with the associated uncertainty as the likelihood-weighted mean and standard deviation of the S_1 models.
6. Save the estimates and the uncertainties on the physical properties along with the fluxes and the physical properties of the best fitting model.
7. Optionally, save the spectrum of the best fit with the individual components (stellar populations, nebular emission, dust emission, etc.), the χ^2 distribution, etc.

Several of the key steps here require further explanation. First of all, as mentioned earlier the SFH is normalised so that its integral is 1 $M_\odot$ (when stellar populations are available, or to 1 W otherwise for the dust emission). This means that in order to obtain the extensive physical properties such as masses or luminosities, we need to rescale the models by a factor α before computing the χ^2. This can be done analytically:

$$\alpha = \frac{\sum_i f_i \times m_i/\sigma_i^2}{\sum_i m_i^2/\sigma_i^2} + \frac{\sum_j f_j \times m_j/\sigma_j^2}{\sum_j m_j^2/\sigma_j^2}, \quad (13)$$

with f_i and m_i being the observed and model fluxes, f_j and m_j the observed and model extensive physical properties, and σ being the corresponding observational uncertainties. Then the computation of the χ^2 is straightforward:

$$\chi^2 = \sum_i \left(\frac{f_i - \alpha \times m_i}{\sigma_i}\right)^2 + \sum_j \left(\frac{f_j - \alpha \times m_j}{\sigma_j}\right)^2 + \sum_k \left(\frac{f_k - m_k}{\sigma_k}\right)^2, \quad (14)$$

with f_k and m_k being the observed and model intensive physical properties. This means that the stellar mass (or the dust luminosity when there is no stellar population included in the model) is not a free parameter even though it is technically possible to treat it as such. This would greatly expand the size of the parameter space by adding an extra dimension, while slowing the

computation of the grid and the analysis by a similar amount, and degrade the accuracy of the estimation of the physical properties.

Optionally, CIGALE can also handle fluxes for which only upper limits have been determined. We adopt the method presented in Appendix A2 of Sawicki (2012). This affects the aforementioned computing steps in several ways. First, the computation of the χ^2 is divided between regular quantities (first three terms corresponding to Eq. (14)) and those with only an upper limit (last three terms):

$$\chi^2 = \sum_i \left(\frac{f_i - \alpha \times m_i}{\sigma_i}\right)^2 + \sum_j \left(\frac{f_j - \alpha \times m_j}{\sigma_j}\right)^2 + \sum_k \left(\frac{f_k - m_k}{\sigma_k}\right)^2$$
$$- 2\sum_i \ln\left(\frac{1}{2}\left[1 + \mathrm{erf}\left(\frac{f_{\mathrm{ul},i} - \alpha \times m_i}{\sqrt{2}\sigma_i}\right)\right]\right)$$
$$- 2\sum_j \ln\left(\frac{1}{2}\left[1 + \mathrm{erf}\left(\frac{f_{\mathrm{ul},j} - \alpha \times m_j}{\sqrt{2}\sigma_j}\right)\right]\right)$$
$$- 2\sum_k \ln\left(\frac{1}{2}\left[1 + \mathrm{erf}\left(\frac{f_{\mathrm{ul},k} - m_k}{\sqrt{2}\sigma_k}\right)\right]\right), \quad (15)$$

with "erf" being the error function, f_{ul} the flux upper limit, and the indices i, j, and k indicating respectively the fluxes, extensive properties, and intensive properties. See Eq. (A6)–(A10) of Sawicki (2012) for a full derivation[6]. The main difficulty is to determine α. For this we have to numerically solve $\partial\chi^2/\partial\alpha = 0$ for every model, which is equivalent to solving Eq. (A11) from Sawicki (2012):

$$\sum_i \left(\frac{f_i - \alpha \times m_i}{\sigma_i}\right) \times \left(\frac{m_i}{\sigma_i}\right) + \sum_j \left(\frac{f_j - \alpha \times m_j}{\sigma_j}\right) \times \left(\frac{m_j}{\sigma_j}\right)$$
$$- \sqrt{\frac{2}{\pi}} \sum_i \frac{m_i \times \exp\left(-\left[(f_{\mathrm{ul},i} - \alpha \times m_i)/\sqrt{2}\sigma_i\right]^2\right)}{\sigma_i \left[1 + \mathrm{erf}\left((f_{\mathrm{ul},i} - \alpha \times m_i)/\sqrt{2}\sigma_i\right)\right]}$$
$$- \sqrt{\frac{2}{\pi}} \sum_j \frac{m_j \times \exp\left(-\left[(f_{\mathrm{ul},j} - \alpha \times m_j)/\sqrt{2}\sigma_j\right]^2\right)}{\sigma_j \left[1 + \mathrm{erf}\left((f_{\mathrm{ul},j} - \alpha \times m_j)/\sqrt{2}\sigma_j\right)\right]} = 0. \quad (16)$$

We do so using the `scipy.optimize.root` root-finding method. Once the χ^2 are computed, the subsequent steps no longer depend on the presence or absence of upper limits.

We have to note that objects in a catalogue do not all need to be fitted with the same set of data. CIGALE will automatically only consider the available data for a given object. The lack of certain data for some targets has a direct impact on some of the aforementioned computation steps. For steps 2 and 3, we simply do not include the data that have been marked as invalid in the input file (value lower than 0 or set to "nan", by convention) in the computation of α and χ^2.

Another feature of this module is the possibility to assess whether or not physical properties can actually be estimated in a reliable way through the analysis of a mock catalogue. The idea is to compare the physical properties of the mock catalogue, which are known exactly, to the estimates from the analysis of the likelihood distribution. To build the mock catalogue we consider the best fit for each object. We then modify each quantity by adding a value taken from a Gaussian distribution with the same standard deviation as the uncertainty on the observation.

---

[6] We should note however that Eq. (A10) from Sawicki (2012), corresponding to Eq. (15) here, contains a mistake, which we have corrected for.

This mock catalogue is then analysed in the exact same way as the original observations. Physical properties for which the exact and estimated values are similar can be estimated reliably.

Applications of the `pdf_analysis` module to a sample of representative galaxies is presented in Sect. 5.3.

5. Examples of CIGALE use cases

As seen above, CIGALE has been designed to be flexible and versatile so that it may have various applications: estimation of the physical properties of an object from the observed SED, generation of theoretical SEDs from analytical considerations or numerical simulations, library to build new tools, or even to serve as a basis for simulating observations. Here we briefly present examples of the former three applications.

5.1. Example of CIGALE as an SED generation tool

The automated generation of SEDs for specific parameters or over different sets of models can be useful for multiple applications: quickly generate artificial observations of galaxies whose physical properties are known (e.g. from numerical simulations), compare observations with grids of models without having to resort to full-scale SED modelling of large samples of galaxies, derive theoretical relations depending on one or more parameters, and so on.

An example of the former use case with CIGALE can be seen in Boquien et al. (2014). They used a set of SFHs from idealised galaxy simulations at $1 \leq z \leq 2$ to simulate observations in star-formation-tracing bands and examine the impact of the SFH on the measure of the SFR. Another example can be found in Álvarez-Márquez et al. (2016), where they defined Lyman-break galaxy selection criteria in the redshift range $2.5 < z < 3.5$ using CIGALE. Here, for the purpose of illustrating the generation of theoretical models, we focus on the latter case. We examine the question of the stellar contribution in the mid-IR. Indeed, even though mid-IR emission in galaxies is often used as a tracer of star-formation (e.g., Calzetti et al. 2007), the stellar contamination can be important. To correct for it the standard method consists in extrapolating the NIR flux to the wavelength of interest (e.g. Helou et al. 2004; Ciesla et al. 2014), exploiting the Rayleigh–Jeans regime of the emission. In practice such methods are calibrated by computing the ratio between near- and mid-IR fluxes, so that with one NIR band that is free of dust, one can estimate the stellar emission in the mid-IR. In Fig. 8 we show how mid-to-near-IR ratios vary depending on the age and timescale of a "delayed" SFH using the Bruzual & Charlot (2003) models assuming a Salpeter (1955) IMF and a metallicity of $Z = 0.02$. The simple configuration file to run this example is shown in Appendix E. It allows to generate a total of 22500 models: the combination of 150 values for the age and 150 values for the timescale τ.

The plots show interesting variations depending on the SFH and the selected NIR and mid-IR wavelengths. Unsurprisingly, when the wavelength difference is small, the variation in the mid-to-near-IR flux ratio is limited to typically less than 20% in the worst cases. In other words, as was noted by Helou et al. (2004), there is only a weak dependence on the SFH. The variation is larger when considering longer wavelengths, such as WISE $22\,\mu$m or MIPS $24\,\mu$m. In this case there can be variations of up to a factor two with a strong dependence on the SFH. The most important difference depends on the age of early-type galaxies (small values for τ), with ~1 Gyr-old galaxies having much bluer colours. When star formation is still on-going (larger values for τ), the colours

show a much weaker dependence on star formation and a constant colour can easily be adopted for late-type galaxies.

This simple example shows how easy it is with CIGALE to explore theoretical grids of models to understand the effects of specific assumptions. We showed here the influence of the age and the timescale of a "delayed" SFH, but similar studies can be done for different parametrisations of the SFH and also with different parameters (IMF, stellar population models, metallicity, presence of dust, presence of an active nucleus, etc.). Beyond the creation of grids of models of a broad range of purposes, the generation of models can also be used in connection to numerical simulations. As mentioned earlier Boquien et al. (2014) used CIGALE to compute the emission of galaxies in star formation tracing bands with respect to time from 23 high-resolution numerical simulations. In a similar way, Ciesla et al. (2015) coupled CIGALE with semi-analytic models to investigate the ability of SED modelling to disentangle the emission of active nuclei and measure their properties.

5.2. Example of CIGALE as a library

The modular and flexible design of CIGALE enables its use not just as a stand-alone package, but also as a library to build new applications well beyond what it was initially conceived for. As a simple example of such a case, we have created a simple pedagogical tool to interactively explore the effect of dust attenuation on the FUV-to-far-IR spectrum of a star-forming galaxy. We show a screen capture of `cigale commander` in Fig. 9. While this example is limited to attenuation curves, it could be extended to all parameters of all modules. Not all applications need to be interactive though. It is also possible to use CIGALE as a database to easily access the base models (single stellar populations, dust emission templates, etc.) in a uniform way without having to write additional code to read the original models that come in different specific formats.

5.3. Example of CIGALE to estimate the physical properties of star-forming galaxies

Measuring the physical properties of galaxies is probably one of the most common uses of SED-modelling codes such as CIGALE and the literature is rich with examples covering a broad range of questions at all redshifts. Various articles have already covered the subject of the reliability of CIGALE to retrieve a number of the intrinsic physical properties of galaxies (e.g. Boquien et al. 2012, 2016; Buat et al. 2014; Lo Faro et al. 2017). As the outcome of such studies naturally depends on the quality and breadth of the data available along with the sampled priors, we refer to these articles for in-depth analyses. Another interesting question is that of how the estimates of the physical properties from different codes compare. To answer this question, Hunt et al. (2019) modelled the SED of the galaxies of the KINGFISH sample (Kennicutt et al. 2011) comparing results from CIGALE, MAGPHYS (da Cunha et al. 2008), and the latest iteration of grasil (Silva et al. 1998). Examining the average SFR over 100 Myr, stellar mass, the FUV attenuation, and the dust luminosity, CIGALE provides excellent consistency with other codes. In another recent effort, this time more geared towards higher redshift samples, CIGALE equally shows excellent performance (Pacifici et al., in prep.).

As a simple illustration of the capabilities of CIGALE, we present the results of the modelling of the star-forming galaxies from the UV to mid-IR SED atlas of Brown et al. (2014). This sample is especially interesting as it provides a set of carefully
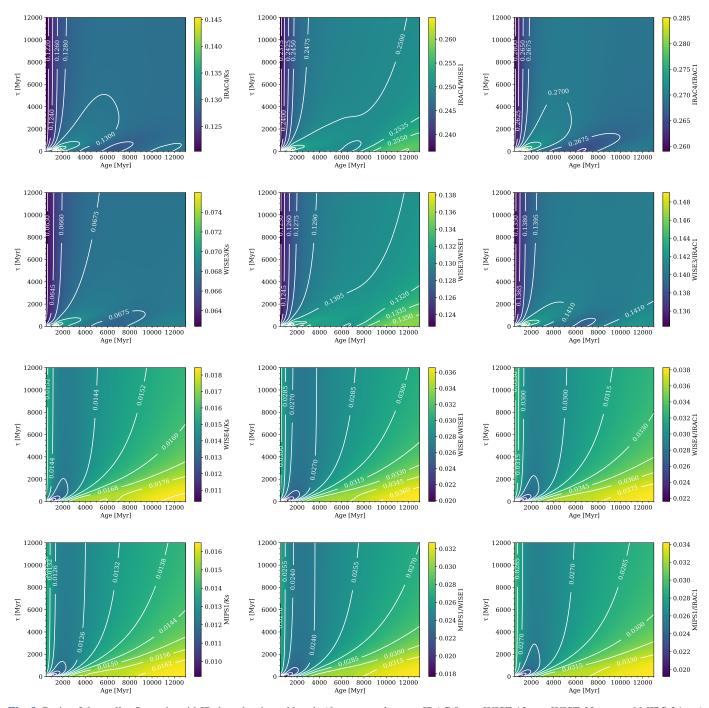
Fig. 8. Ratio of the stellar fluxes in mid-IR dust-dominated bands (*from top to bottom*: IRAC 8 μm, WISE 12 μm, WISE 22 μm, and MIPS 24 μm) and NIR dust-free bands (*from left to right*: Ks 2.2 μm, WISE 3.4 μm, and IRAC 3.6 μm) bands depending on the age of the galaxy (*x*-axis) and the τ constant of a "delayed" SFH (*y*-axis). The colour indicates the value of the ratio of the fluxes according to the colour bar to the right of each plot, with white lines indicating isocontours. A total of 22 500 models were computed representing the combination of 150 parameters on each axis. The Bruzual & Charlot (2003) models were adopted, assuming a Salpeter (1955) IMF and a metallicity $Z = 0.02$.

vetted photometric data, which is ideal for SED modelling. We select a subsample of 78 star-forming galaxies which 1) have both FUV and NUV fluxes, and 2) are classified either as Sa or later-type or as peculiar. We exclude galaxies that are classified as purely AGN by Brown et al. (2014) as this may contaminate in an appreciable way both the UV and/or the mid-IR ends of the spectra, significantly affecting the results[7]. We model these

galaxies with a representative set of modules as described in Table 1, leading to a modest grid of 8164800 models. An example of a typical best fit is shown in Fig. 10.

While `CIGALE` can provide measurements for numerous physical properties, we chose here to concentrate on a subsample of six quantities that are frequently used to study galaxies: the FUV attenuation ($A_{\rm FUV}$), the dust luminosity ($L_{\rm dust}$), the instantaneous SFR, the stellar mass ($M_{\star}$), the UV slope (β, Calzetti et al. 1994), and the decimal logarithm of the ratio between the IR and FUV luminosities (IRX). The distributions of these six physical properties are shown in Fig. 11.

---

[7] For simplicity here we do not include AGN models. We refer to Ciesla et al. (2015) for a presentation of the capabilities of `CIGALE` regarding AGNs.
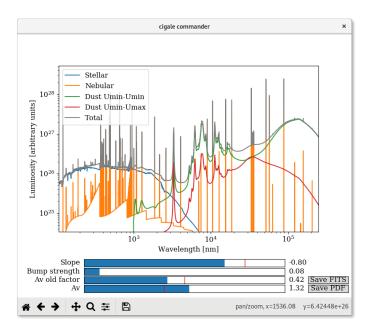
Fig. 9. Screenshot of a simple application built using `CIGALE` as a library. It is designed to interactively explore the effect of dust attenuation on the FUV-to-far-IR spectrum of a star-forming galaxy. Here the key parameters for the attenuation can easily be changed with sliders, allowing for a rapid examination of the impact of each parameter. The script is approximately only 150 lines long. It transparently uses `CIGALE` modules to build the SED from user-provided parameters and `matplotlib` to display the plot and handle the sliders.

Even though `CIGALE` will always provide an estimate of the physical properties, this estimate may or may not be reliable depending on the wavelength coverage, the quality of the data, and the explored parameter space, among other factors. A standard way to test whether the physical properties can at least be retrieved in a self-consistent way is through a mock catalogue as described in Sect. 4.3. As a reminder, in a nutshell the idea is to fit the observations and build an artificial catalogue from the best fits. Considering these best fits, we know exactly what the corresponding physical parameters are, so we know the "truth". Noise is then injected into the fluxes of this new catalogue to simulate new observations. Fitting these artificial observations we can then compare the inferred physical properties from the likelihood distribution to their actual values. We show such an analysis in Fig. 12 for the six aforementioned physical properties.

Overall, all the physical quantities are reliably estimated with the average estimated values being remarkably close to the true values. Regarding the scatter around this mean value, there is a marked difference in performance. The SFR, L_{dust}, and IRX show very little scatter relative to the true values, lower than 0.1 dex (IRX is already a log-scale quantity). The two quantities with the lower performance appear to be β (scatter of 0.217 or less than 5% of the dynamical range) and A(FUV) (scatter of 0.157 dex). In either case, this performance remains excellent. This slight difference in performance reflects the fact that not all physical quantities can be determined with the same reliability, partly due to intrinsic degeneracies between the physical processes at play, and partly due to the breadth and quality of the photometric coverage. It is important to note that even though this exercise is useful, it does not take into account the uncertainties due to the reliability of the models themselves, and therefore yields lower limits on the actual uncertainties on the physical properties.
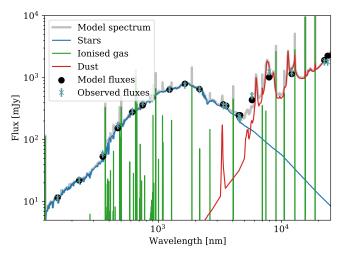
Fig. 10. Best-fitting model for the spiral galaxy NGC 4321 (grey) located in the nearby Virgo cluster, showing the total stellar (blue, the dust attenuation has already been accounted for), nebular (green, also including dust attenuation), and dust (red) emission. The model fluxes in passbands computed using Eq. (10) are indicated with black circles. These fluxes were then fitted to the observations (turquoise cross with the uncertainties indicated with the vertical lines), yielding a final reduced $\chi^2 \simeq 0.5$.

With this inspection done, well-measured physical quantities can then be used to understand the properties of these galaxies. As a very simple example, we plot here the classical relation between IRX and the UV spectral slope, β, in Fig. 13. This relation is important as β is often used to estimate the attenuation of distant galaxies where dust observations are missing, and IRX is a simple proxy for the attenuation. Numerous works have found and tried to explain the extensive variations observed in the IRX-β relation over the years (e.g. Kong et al. 2004; Burgarella et al. 2005; Boquien et al. 2009, 2012; Overzier et al. 2011; Grasha et al. 2013; Ye et al. 2016; Popping et al. 2017; Reddy et al. 2018, and many others). This simple example shows the effect of the SFH to explain why more quiescent galaxies deviate from the relation followed by starburst galaxies, with more active galaxies following the relation of Kong et al. (2004) and more quiescent galaxies progressively moving closer to the relation of Boquien et al. (2012). It is possible to investigate such questions, and many more, thanks to the flexibility of `CIGALE` in modelling a broad range of galaxies from intense starburst to elliptical galaxies and estimating their physical properties.

6. Summary

In this paper, we present the new generation of the Code Investigating GALaxy Emission, `CIGALE`. Three principles have guided its development: modularity (it is easy to add new modules, and we encourage and support such developments, or swap modules modelling the same component), clarity (the code is easy to use and to understand), and efficiency (it runs quickly and is parallelised to take advantage of modern processors with multiple cores), both for the developers and for the users.

In practical terms, CIGALE is based on an energy balance principle (the energy absorbed by dust in the UV-to-near-IR domain is re-emitted self-consistently in the mid- and far-IR). The models from the FUV to the radio are built in a modular way, taking into account flexible SFH and stellar populations (Bruzual & Charlot 2003; Maraston 2005), ionised

Table 1. Modules and parameter values used to model the sample of Brown et al. (2014).

Module	Parameter	Value
sfhdelayed	tau_main (10^6 years)	1, 500, 1000, 2000, 3000, 4000, 5000, 6000, 7000, 8000
	age_main (10^6 years)	13 000
	tau_burst (10^6 years)	10^9
	age_burst (10^6 years)	5, 10, 25, 50, 100, 200, 350, 500, 750, 1000
	f_burst	0, 0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1, 0.25
bc03	imf	1 (Chabrier)
	metallicity	0.02
nebular	logU	−3.0
	f_esc	0.0
	f_dust	0.0
	lines_width (km s^{-1})	300
dustatt_modified_starburst	E_BV_nebular (mag)	0.005, 0.01, 0.025, 0.05, 0.075, 0.10, 0.15, 0.20, 0.25, 0.30, 0.35, 0.40, 0.45, 0.50, 0.55, 0.60
	E_BV_factor	0.25, 0.50, 0.75
	uv_bump_wavelength (nm)	217.5
	uv_bump_width (nm)	35.0
	uv_bump_amplitude	0.0, 1.5, 3.0 (Milky Way)
	powerlaw_slope	−0.5, −0.4, −0.3, −0.2, −0.1, 0.0
	Ext_law_emission_lines	1 (Milky Way)
	Rv	3.1
	filters	FUV, V_B90
dale2014	alpha	0.5, 1.0, 1.5, 2.0, 3.5, 3.0, 3.5, 4.0
restframe_parameters	beta_calz94	True
	D4000	False
	IRX	True
	EW_lines	500.7/1.0 & 656.3/1.0
	luminosity_filters	FUV & V_B90
	colours_filters	FUV-NUV & NUV-r_prime
redshifting	redshift	0

Notes. The grid of models (fluxes and physical properties) is estimated over all the possible combinations of parameters, leading to a total of 8164800 models. The corresponding `pcigale.ini` file is provided in Appendix E.
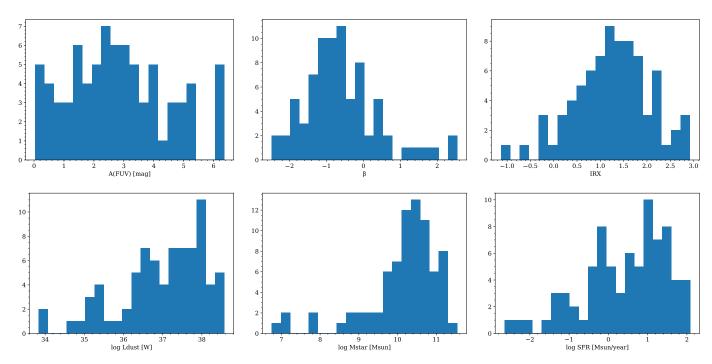
Fig. 11. Distribution of the physical properties of the Brown et al. (2014) sample. We see that in a single run CIGALE can model galaxies with a wide range of properties: FUV attenuation, UV slope, IR-to-UV luminosity, dust luminosity, stellar mass, and SFR. This is only a small excerpt of the numerous physical properties that can also be estimated with CIGALE (see Appendix C).
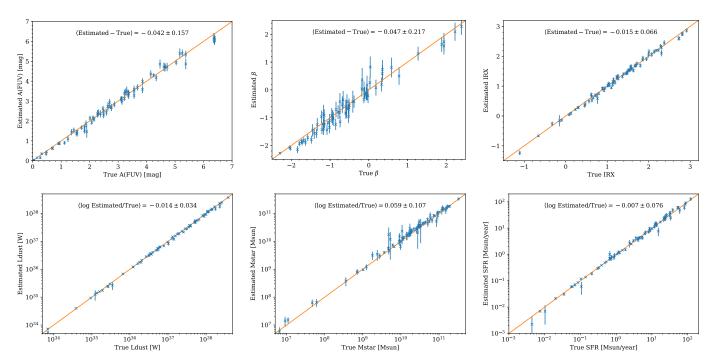
Fig. 12. Comparison between the true (x-axis) and estimated (y-axis) values of A_{FUV}, β, IRX, L_{dust}, $M_\star$, and the SFR from the upper-left to the lower-right panel. Each galaxy is indicated with a blue cross, with the vertical line giving the 1-σ error bar on the estimate. From this analysis it is clear that with this dataset and parameter space CIGALE can measure these physical properties self-consistently. Not all physical properties are equally well measured however. It is apparent for instance that if the dust luminosity is extremely reliable (-0.014 ± 0.034 dex), the performance for β for example, even though still excellent, shows somewhat more dispersion (-0.047 ± 0.217 dex).
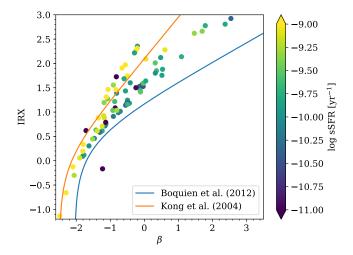
Fig. 13. Relation between the observed IR-to-FUV luminosity (IRX) and the UV spectral slope (β) for the 129 galaxies of the Brown et al. (2014) sample. The colour of each symbol indicates the specific SFR, following the scale given by the colour bar to the right. The locus followed by resolved quiescent star-forming (respectively starburst) galaxies from Boquien et al. (2012; respectively Kong et al. 2004) is indicated by the blue (respectively orange) line.

gas (Inoue 2011), attenuation by dust (Calzetti et al. 2000; Charlot & Fall 2000) and re-emission of the energy at longer wavelengths (Draine & Li 2007; Casey 2012; Dale et al. 2014), active nuclei (Fritz et al. 2006; Dale et al. 2014), and the IGM (Meiksin 2006). The computation of these models is done in a parallel way on grids of models that can reach several hundred million elements. These models can then be simply saved for theoretical studies or can be used to evaluate a wide range of physical properties for observed objects. Such evaluation is

based on the likelihood-weighted mean and standard deviation, taking into account the presence of upper limits. Finally, thanks to its versatility, CIGALE can also be used as a library to build new applications.

This article has presented a snapshot of the current state of CIGALE. It is however a constantly evolving code and we will present in upcoming papers new major evolutions to adapt it to the ever changing challenges of panchromatic modelling. The code along with its documentation is publicly available at http://cigale.lam.fr.

References

Álvarez-Márquez, J., Burgarella, D., Heinis, S., et al. 2016, A&A, 587, A122

Anders, P., & Fritze-v. Alvensleben, U. 2003, A&A, 401, 1063

Astropy Collaboration (Robitaille, T. P., et al.) 2013, A&A, 558, A33

Astropy Collaboration (Price-Whelan, A. M., et al.) 2018, AJ, 156, 123

Balogh, M. L., Morris, S. L., Yee, H. K. C., Carlberg, R. G., & Ellingson, E. 1999, ApJ, 527, 54

Bianchi, S. 2008, A&A, 490, 461

Bitsakis, T., Dultzin, D., Ciesla, L., et al. 2016, MNRAS, 459, 957

Boissier, S., Prantzos, N., Boselli, A., & Gavazzi, G. 2003, MNRAS, 346, 1215

Boquien, M., Duc, P., Wu, Y., et al. 2009, AJ, 137, 4561

Boquien, M., Duc, P., Galliano, F., et al. 2010, AJ, 140, 2124

Boquien, M., Buat, V., Boselli, A., et al. 2012, A&A, 539, A145
Boquien, M., Boselli, A., Buat, V., et al. 2013, A&A, 554, A14
Boquien, M., Buat, V., & Perret, V. 2014, A&A, 571, A72
Boquien, M., Kennicutt, R., Calzetti, D., et al. 2016, A&A, 591, A6
Brown, M. J. I., Moustakas, J., Smith, J.-D. T., et al. 2014, ApJS, 212, 18
Bruzual, G., & Charlot, S. 2003, MNRAS, 344, 1000
Buat, V., Boissier, S., Burgarella, D., et al. 2008, A&A, 483, 107
Buat, V., Giovannoli, E., Heinis, S., et al. 2011, A&A, 533, A93
Buat, V., Noll, S., Burgarella, D., et al. 2012, A&A, 545, A141
Buat, V., Heinis, S., Boquien, M., et al. 2014, A&A, 561, A39
Buat, V., Boquien, M., Małek, K., et al. 2018, A&A, 619, A135
Burgarella, D., Buat, V., & Iglesias-Páramo, J. 2005, MNRAS, 360, 1413
Burgarella, D., Heinis, S., Magdis, G., et al. 2011, ApJ, 734, L12
Calzetti, D., Kinney, A. L., & Storchi-Bergmann, T. 1994, ApJ, 429, 582
Calzetti, D., Armus, L., Bohlin, R. C., et al. 2000, ApJ, 533, 682
Calzetti, D., Kennicutt, R. C., Engelbracht, C. W., et al. 2007, ApJ, 666, 870
Cardelli, J. A., Clayton, G. C., & Mathis, J. S. 1989, ApJ, 345, 245
Casey, C. M. 2012, MNRAS, 425, 3094
Chabrier, G. 2003, PASP, 115, 763
Chan, T. F., Golub, G. H., & LeVeque, R. J. 1979, Technical Report STAN-CS-79-773, Department of Computer Science, Stanford University
Charlot, S., & Fall, S. M. 2000, ApJ, 539, 718
Ciesla, L., Boquien, M., Boselli, A., et al. 2014, A&A, 565, A128
Ciesla, L., Charmandaris, V., Georgakakis, A., et al. 2015, A&A, 576, A10
Ciesla, L., Boselli, A., Elbaz, D., et al. 2016, A&A, 585, A43
Ciesla, L., Elbaz, D., & Fensch, J. 2017, A&A, 608, A41
Conroy, C., & Gunn, J. E. 2010, ApJ, 712, 833
Conroy, C., Gunn, J. E., & White, M. 2009, ApJ, 699, 486
da Cunha, E., Charlot, S., & Elbaz, D. 2008, MNRAS, 388, 1595
Dale, D. A., & Helou, G. 2002, ApJ, 576, 159
Dale, D. A., Helou, G., Magdis, G. E., et al. 2014, ApJ, 784, 83
de Barros, S., Schaerer, D., & Stark, D. P. 2014, A&A, 563, A81
de Looze, I., Baes, M., Bendo, G. J., et al. 2012, MNRAS, 427, 2797
De Looze, I., Fritz, J., Baes, M., et al. 2014, A&A, 571, A69
Draine, B. T., & Li, A. 2007, ApJ, 657, 810
Draine, B. T., Aniano, G., Krause, O., et al. 2014, ApJ, 780, 172
Engelbracht, C. W., Gordon, K. D., Rieke, G. H., et al. 2005, ApJ, 628, L29
Ferland, G. J. 1980, PASP, 92, 596
Ferland, G. J., Korista, K. T., Verner, D. A., et al. 1998, PASP, 110, 761
Ferland, G. J., Porter, R. L., van Hoof, P. A. M., et al. 2013, Rev. Mex. Astron. Astrofis., 49, 137
Fritz, J., Franceschini, A., & Hatziminaoglou, E. 2006, MNRAS, 366, 767
Giovannoli, E., Buat, V., Noll, S., Burgarella, D., & Magnelli, B. 2011, A&A, 525, A150
Gordon, K. D., Misselt, K. A., Witt, A. N., & Clayton, G. C. 2001, ApJ, 551, 269
Grasha, K., Calzetti, D., Andrews, J. E., Lee, J. C., & Dale, D. A. 2013, ApJ, 773, 174
Hayes, M., Schaerer, D., Östlin, G., et al. 2011, ApJ, 730, 8
Helou, G., Soifer, B. T., & Rowan-Robinson, M. 1985, ApJ, 298, L7
Helou, G., Roussel, H., Appleton, P., et al. 2004, ApJS, 154, 253
Hirashita, H., Burgarella, D., & Bouwens, R. J. 2017, MNRAS, 472, 4587
Hunt, L. K., De Looze, I., Boquien, M., et al. 2019, A&A, 621, A51
Hunter, J. D. 2007, Comput. Sci. Eng., 9, 90
Inoue, A. K. 2001, AJ, 122, 1788

Inoue, A. K. 2010, MNRAS, 401, 1325
Inoue, A. K. 2011, MNRAS, 415, 2920
Inoue, A. K., Hirashita, H., & Kamaya, H. 2001, ApJ, 555, 613
Inoue, A. K., Iwata, I., & Deharveng, J.-M. 2006, MNRAS, 371, L1
Johnston, R., Vaccari, M., Jarvis, M., et al. 2015, MNRAS, 453, 2540
Jones, A. P., Köhler, M., Ysard, N., Bocchio, M., & Verstraete, L. 2017, A&A, 602, A46
Jones, E., Oliphant, T., Peterson, P., et al. 2001, SciPy: Open source scientific tools for Python
Kennicutt, R. C., Calzetti, D., Aniano, G., et al. 2011, PASP, 123, 1347
Kong, X., Charlot, S., Brinchmann, J., & Fall, S. M. 2004, MNRAS, 349, 769
Koornneef, J., Bohlin, R., Buser, R., Horne, K., & Turnshek, D. 1986, Highlights Astron., 7, 833
Kroupa, P. 2001, MNRAS, 322, 231
Leitherer, C., Li, I.-H., Calzetti, D., & Heckman, T. M. 2002, ApJS, 140, 303
Lo Faro, B., Buat, V., Roehlly, Y., et al. 2017, MNRAS, 472, 1372
Małek, K., Pollo, A., Takeuchi, T. T., et al. 2014, A&A, 562, A15
Małek, K., Buat, V., Roehlly, Y., et al. 2018, A&A, 620, A50
Maraston, C. 2005, MNRAS, 362, 799
Meiksin, A. 2006, MNRAS, 365, 807
Nagao, T., Maiolino, R., Marconi, A., & Matsuhara, H. 2011, A&A, 526, A149
Noll, S., Burgarella, D., Giovannoli, E., et al. 2009, A&A, 507, 1793
O'Donnell, J. E. 1994, ApJ, 422, 158
Oliphant, T. E. 2007, Comput. Sci. Eng., 9, 10
Overzier, R. A., Heckman, T. M., Wang, J., et al. 2011, ApJ, 726, L7
Pacifici, C., Charlot, S., Blaizot, J., & Brinchmann, J. 2012, MNRAS, 421, 2002
Papovich, C., Dickinson, M., & Ferguson, H. C. 2001, ApJ, 559, 620
Pappalardo, C., Bizzocchi, L., Fritz, J., et al. 2016, A&A, 589, A11
Pei, Y. C. 1992, ApJ, 395, 130
Popescu, C. C., Misiriotis, A., Kylafis, N. D., Tuffs, R. J., & Fischera, J. 2000, A&A, 362, 138
Popping, G., Puglisi, A., & Norman, C. A. 2017, MNRAS, 472, 2315
Reddy, N. A., Kriek, M., Shapley, A. E., et al. 2015, ApJ, 806, 259
Reddy, N. A., Steidel, C. C., Pettini, M., & Bogosavljević, M. 2016, ApJ, 828, 107
Reddy, N. A., Oesch, P. A., Bouwens, R. J., et al. 2018, ApJ, 853, 56
Salim, S., Boquien, M., & Lee, J. C. 2018, ApJ, 859, 11
Salpeter, E. E. 1955, ApJ, 121, 161
Sandage, A. 1986, A&A, 161, 89
Sawicki, M. 2012, PASP, 124, 1208
Silva, L., Granato, G. L., Bressan, A., & Danese, L. 1998, ApJ, 509, 103
Stark, D. P., Schenker, M. A., Ellis, R., et al. 2013, ApJ, 763, 129
Trayford, J. W., Camps, P., Theuns, T., et al. 2017, MNRAS, 470, 771
Tuffs, R. J., Popescu, C. C., Völk, H. J., Kylafis, N. D., & Dopita, M. A. 2004, A&A, 419, 821
Viaene, S., Baes, M., Tamm, A., et al. 2017, A&A, 599, A64
Vika, M., Ciesla, L., Charmandaris, V., Xilouris, E. M., & Lebouteiller, V. 2017, A&A, 597, A51
Wild, V., Charlot, S., Brinchmann, J., et al. 2011, MNRAS, 417, 1760
Worthey, G. 1994, ApJS, 95, 107
Xilouris, E. M., Byun, Y. I., Kylafis, N. D., Paleologou, E. V., & Papamastorakis, J. 1999, A&A, 344, 868
Ye, C., Zou, H., Lin, L., et al. 2016, ApJ, 826, 209

Appendix A: General workflow

A general schematic of `CIGALE`, including the important choices detailed in Appendix B, is shown in Fig. A.1.

Fig. A.1. General workflow of `CIGALE` for SED fitting and parameter estimation. It can be split into 4 broad consecutive steps: 1) input handling in which configuration and data files are processed, 2) model generation in which the model SEDs and the associated physical properties are computed and stored in memory, 3) analysis in which the models are fitted to the observations and physical properties are estimated through the PDF, and finally 4) output handling in which the results are saved to disk. The input and output steps are shown in green, the parallel computation of the models and of the fits are shown in blue, and the main data residing in shared memory are shown in red.

Appendix B: Important implementation choices

The algorithms described above are central for computing models and estimating the physical properties of galaxies. However, applied as is, they would lead to extremely poor performance. We describe here some of the implementation choices we have made in `CIGALE` to achieve high performance with limited resources: parallelisation, caching, and computation by blocks of models.

B.1. Parallelisation strategy

With the increasing number of cores available in off-the-shelf computers, it is important that `CIGALE` can make full use of them. We have therefore parallelised the model generation and the analysis parts of `pcigale` as well as the generation of plots with `pcigale-plots`. The implementation makes use of the `multiprocessing` module of `python`, which allows for a process-based parallelism. The strategy is to create an arbitrary number of child processes (in general, but this is user-defined, corresponding to the number of available cores), each computing one model or analysing one object at a time, on-demand from the main process.

With this strategy, costly communications between the main process and individual sub-processes can prove to be a bottleneck. To limit this issue, the computed models fluxes and physical properties are written into shared arrays accessible from all sub-processes rather than being communicated back to the main process. These shared arrays are also accessed directly by each sub-process carrying out the analysis of the objects and results are written into another set of shared arrays. This optimises both the computation speed and memory usage as no copy of the shared arrays needs to be made for individual sub-processes.

While launching the sub-processes incurs some slight start-up delay, the computation speed increase initially grows with the number of physical cores. However, the communication between the memory and the processor progressively becomes the main bottleneck with increasing numbers of cores, yielding sub-linear gains. Finally, while the gain of this method is substantial, there is however a small memory cost from using sub-processes.

B.2. Cache system

Another important aspect of the implementation of `CIGALE` to minimise computing time is the cache system. Caching is done at two levels. We cache the model creation modules and we cache partially computed SEDs.

The SED of an object is built by adding different physical components, and each of these physical components is added by a specific module, which is fully determined by its input parameters. Rather than instantiating repeatedly the same module each time we need to apply it to an SED, we instantiate it only once and store it in cache. This is especially valuable because the initialisation of a module can be expensive as it loads and computes all the data that do not depend on other modules. With caching, rather than being carried out each time we apply this module to an SED, the computations are done only once for the entire run.

We also store intermediate SED in cache to avoid having to compute each model from scratch. For instance the cache can contain an SED built with steps 1–6 (that is the full model but without redshift); steps 7 and 8 can then easily be applied to compute the flux at different redshifts without computing steps 1–6 again. Because SED objects can be particularly voluminous, they contain the full spectrum for each of the physical components, intermediate SED that are no longer of use to construct new SEDs are discarded. This is the case for instance for an SED that corresponds to an SFH not used by any of the models still to be computed. The models are computed in an order designed to optimise the efficiency of the cache. In practice at any given time it only contains a small number of intermediate SEDs necessary to ensure the full speedup, maximising cache lookup speed and minimising memory usage.

B.3. Computation by blocks of models

As detailed in Sect. 4, the grid of models is computed in one step. While there are advantages in having all the models in memory (in particular in terms of speed), this can be very demanding for large grids. To allow the computation of grids of models larger than the size of the computer memory, `CIGALE` can compute models in several blocks. After the computation of each block, the physical properties of the targets are estimated using exclusively the models from this block and the results (likelihood-weighted means, likelihood-weighted standard deviations, best fit, sum of the likelihoods) stored in memory. After the physical properties have been estimated for all the blocks, the results are combined to determine the overall likelihood-weighted means and standard deviations, and the overall best fit. If finding the overall best fit is trivial, as it is simply of lowest χ^2 considering all the blocks, the estimation of the likelihood weighted physical properties is more complex. To do so, we adopt the variance parallel algorithm presented in Chan et al. (1979), where we replace the number of elements in each block with the sum of the likelihood. Let w_i be the sum of the likelihoods and m_i the likelihood-weighted mean of a physical property for the block of models j. The overall likelihood-weighted mean m is:

$$m = \frac{\sum_i m_i \times w_i}{\sum_i w_i}.$$ (B.1)

The computation of the overall likelihood-weighted standard deviation σ_i from the likelihood-weighted standard deviations for individual blocks σ_i is more involved:

$$\sigma^2 = \sum_i \sigma_i^2 \times w_i + \frac{\sum_j (m_{i+1} - m_i)^2 / \sum_i w_i}{\sum_i w_i}.$$ (B.2)

Appendix C: Input and output physical properties

C.1. Star formation history

C.1.1. `sfh2exp`

Table C.1. Input parameters of the `sfh2exp` module.

Parameter	Unit	Description
tau_main	Myr	e-folding time of the main stellar population model
tau_burst	Myr	e-folding time of the late starburst population model
f_burst	–	Mass fraction of the late burst population
age	Myr	Age of the main stellar population in the galaxy
burst_age	Myr	Age of the late burst
sfr_0	$M_\odot$ yr^{-1}	Multiplicative factor controlling the amplitude of SFR if normalise is False
normalise	–	Normalise the SFH to produce one solar mass

Table C.2. Output parameters of the `sfh2exp` module.

Parameter	Unit	Description
sfh.sfr	$M_\odot$ yr^{-1}	Instantaneous SFR
sfh.sfr10 Myrs	$M_\odot$ yr^{-1}	Average SFR over 10 Myr
sfh.sfr100 Myrs	$M_\odot$ yr^{-1}	Average SFR over 100 Myr
sfh.age	Myr	Age of the main stellar population in the galaxy
sfh.integrated	$M_\odot$	Integral of the SFH
sfh.tau_main	Myr	e-folding time of the main stellar population model
sfh.tau_burst	Myr	e-folding time of the late starburst population model
sfh.f_burst	–	Mass fraction of the late burst population
sfh.burst_age	Myr	Age of the late burst in Myr

C.1.2. `sfhdelayed`

Table C.3. Input parameters of the `sfhdelayed` module.

Parameter	Unit	Description
tau_main	Myr	e-folding time of the main stellar population model
age_main	Myr	Age of the main stellar population in the galaxy
tau_burst	Myr	e-folding time of the late starburst population model
age_burst	Myr	Age of the late burst
f_burst	–	Mass fraction of the late burst population
sfr_A	$M_\odot\,\mathrm{yr}^{-1}$	Multiplicative factor controlling the amplitude of SFR if normalise is False
normalise	–	Normalise the SFH to produce one solar mass

Table C.4. Output parameters of the `sfhdelayed` module.

Parameter	Unit	Description
sfh.sfr	$M_\odot\,\mathrm{yr}^{-1}$	Instantaneous SFR
sfh.sfr10 Myrs	$M_\odot\,\mathrm{yr}^{-1}$	Average SFR over 10 Myr
sfh.sfr100 Myrs	$M_\odot\,\mathrm{yr}^{-1}$	Average SFR over 100 Myr
sfh.integrated	$M_\odot$	Integral of the SFH
sfh.age_main	Myr	Age of the main stellar population in the galaxy
sfh.tau_main	Myr	e-folding time of the main stellar population model
sfh.age_burst	Myr	Age of the late burst
sfh.tau_burst	–	e-folding time of the late starburst population model
sfh.f_burst	Myr	Mass fraction of the late burst population

C.1.3. `sfhdelayedbq`

Table C.5. Input parameters of the `sfhdelayedbq` module.

Parameter	Unit	Description
tau_main	Myr	e-folding time of the main stellar population model
age_main	Myr	Age of the main stellar population in the galaxy
age_bq	Myr	Age of the burst/quench
r_sfr	–	Ratio of the SFR after/before age_bq
sfr_A	$M_\odot\,\mathrm{yr}^{-1}$	Value of SFR at $t = 0$ if normalise is False
normalise	–	Normalise the SFH to produce one solar mass

Table C.6. Output parameters of the `sfhdelayedbq` module.

Parameter	Unit	Description
sfh.sfr	$M_\odot\,\mathrm{yr}^{-1}$	Instantaneous SFR
sfh.sfr10 Myrs	$M_\odot\,\mathrm{yr}^{-1}$	Average SFR over 10 Myr
sfh.sfr100 Myrs	$M_\odot\,\mathrm{yr}^{-1}$	Average SFR over 100 Myr
sfh.integrated	$M_\odot$	Integral of the SFH
sfh.age_main	Myr	Age of the main stellar population in the galaxy
sfh.tau_main	Myr	e-folding time of the main stellar population model
sfh.age_bq	Myr	Age of the burst/quench
sfh.r_sfr	–	Ratio of the SFR after/before age_bq

C.1.4. sfhperiodic

Table C.7. Input parameters of the `sfhperiodic` module.

Parameter	Unit	Description
type_bursts	–	Type of the individual star formation episodes. 0: exponential, 1: delayed, 2: rectangle
delta_bursts	Myr	Elapsed time between the beginning of each burst
tau_bursts	Myr	Duration (rectangle) or e-folding time of all short events
age	Myr	Age of the main stellar population in the galaxy
sfr_A	$M_\odot\,\mathrm{yr}^{-1}$	Multiplicative factor controlling the amplitude of SFR (valid for each event) if normalise is False
normalise	–	Normalise the SFH to produce one solar mass

Table C.8. Output parameters of the `sfhperiodic` module.

Parameter	Unit	Description
sfh.sfr	$M_\odot\,\mathrm{yr}^{-1}$	Instantaneous SFR
sfh.sfr10 Myrs	$M_\odot\,\mathrm{yr}^{-1}$	Average SFR over 10 Myr
sfh.sfr100 Myrs	$M_\odot\,\mathrm{yr}^{-1}$	Average SFR over 100 Myr
sfh.integrated	$M_\odot$	Integral of the SFH
sfh.type_bursts	–	Type of the individual star formation episodes
sfh.delta_bursts	Myr	Elapsed time between the beginning of each burst
sfh.tau_bursts	Myr	Duration (rectangle) or e-folding time of all short events

C.1.5. sfh_buat08

Table C.9. Input parameters of the `sfh_buat08` module.

Parameter	Unit	Description
velocity	$\mathrm{km\,s}^{-1}$	Rotational velocity of the galaxy
age	Myr	Age of the oldest stars in the galaxy
normalise	–	Normalise the SFH to produce one solar mass

Table C.10. Output parameters of the `sfh_buat08` module.

Parameter	Unit	Description
sfh.sfr	$M_\odot\,\mathrm{yr}^{-1}$	Instantaneous SFR
sfh.sfr10 Myrs	$M_\odot\,\mathrm{yr}^{-1}$	Average SFR over 10 Myr
sfh.sfr100 Myrs	$M_\odot\,\mathrm{yr}^{-1}$	Average SFR over 100 Myr
sfh.integrated	$M_\odot$	Integral of the SFH
sfh.velocity	$\mathrm{km\,s}^{-1}$	Rotational velocity of the galaxy

C.1.6. sfhfromfile

Table C.11. Input parameters of the `sfhfromfile` module.

Parameter	Unit	Description
filename	–	Name of the file containing the SFH
sfr_column	–	List of column indices of the SFR
age	Myr	Age in Myr at which the SFH will be looked at
normalise	–	Normalise the SFH to produce one solar mass

Table C.12. Output parameters of the `sfhfromfile` module.

Parameter	Unit	Description
sfh.sfr	$M_\odot \, \mathrm{yr}^{-1}$	Instantaneous SFR
sfh.sfr10 Myrs	$M_\odot \, \mathrm{yr}^{-1}$	Average SFR over 10 Myr
sfh.sfr100 Myrs	$M_\odot \, \mathrm{yr}^{-1}$	Average SFR over 100 Myr
sfh.integrated	$M_\odot$	Integral of the SFH
sfh.index	–	Index of the column

C.2. Stellar populations

C.2.1. `bc03`

Table C.13. Input parameters of the `bc03` module.

Parameter	Unit	Description
imf	–	Initial mass function: 0 (Salpeter) or 1 (Chabrier)
metallicity	–	Metallicity
separation_age	Myr	Age of the separation between the young and the old star populations

Table C.14. Output parameters of the `bc03` module.

Parameter	Unit	Description
stellar.imf	–	Initial mass function: 0 (Salpeter) or 1 (Chabrier)
stellar.metallicity	–	Metallicity
stellar.old_young_separation_age	Myr	Age of the separation between the young and the old star populations
stellar.m_star_young	$M_\odot$	Stellar mass of the young population
stellar.m_gas_young	$M_\odot$	Gas mass of the young population
stellar.n_ly_young	Photons	Number of Lyman continuum photons of the young population
stellar.lum_ly_young	W	Luminosity of the Lyman continuum of the young population
stellar.lum_young	W	Luminosity of the young population
stellar.m_star_old	$M_\odot$	Stellar mass of the old population
stellar.m_gas_old	$M_\odot$	Gas mass of the old population
stellar.n_ly_old	Photons	Number of Lyman continuum photons of the old population
stellar.lum_ly_old	W	Luminosity of the Lyman continuum of the old population
stellar.lum_old	W	Luminosity of the old population
stellar.m_star	$M_\odot$	Total stellar mass
stellar.m_gas	$M_\odot$	Total gas mass
stellar.n_ly	Photons	Total number of Lyman continuum photons
stellar.lum_ly	W	Total Lyman continuum luminosity
stellar.lum	W	Total luminosity
stellar.age_m_star	Myr	Mass-weighted age

C.2.2. `m2005`

Table C.15. Input parameters of the `m2005` module.

Parameter	Unit	Description
imf	–	Initial mass function: 0 (Salpeter) or 1 (Chabrier)
metallicity	–	Metallicity
separation_age	Myr	Age of the separation between the young and the old star populations

Table C.16. Output parameters of the `m2005` module.

Parameter	Unit	Description
stellar.imf	–	Initial mass function: 0 (Salpeter) or 1 (Chabrier)
stellar.metallicity	–	Metallicity
stellar.old_young_separation_age	Myr	Age of the separation between the young and the old star populations
stellar.mass_total_young	$M_\odot$	Stellar mass of the young population
stellar.mass_alive_young	$M_\odot$	Alive stars mass of the young population
stellar.mass_white_dwarf_young	$M_\odot$	White dwarfs mass of the young population
stellar.mass_neutron_young	$M_\odot$	Neutron stars mass of the young population
stellar.mass_black_hole_young	$M_\odot$	Black hole mass of the young population
stellar.lum_old	W	Luminosity of the old population
stellar.mass_total_old	$M_\odot$	Stellar mass of the old population
stellar.mass_alive_old	$M_\odot$	Alive stars mass of the old population
stellar.mass_white_dwarf_old	$M_\odot$	White dwarfs mass of the old population
stellar.mass_neutron_old	$M_\odot$	Neutron stars mass of the old population
stellar.mass_black_hole_old	$M_\odot$	Black hole mass of the old population
stellar.lum_old	W	Luminosity of the old population
stellar.mass_total	$M_\odot$	Total stellar mass
stellar.mass_alive	$M_\odot$	Total alive stars mass
stellar.mass_white_dwarf	$M_\odot$	Total white dwarfs mass
stellar.mass_neutron	$M_\odot$	Total neutron stars mass
stellar.mass_black_hole	$M_\odot$	Total black hole mass
stellar.lum	W	Total luminosity

C.3. Nebular emission

C.3.1. nebular

Table C.17. Input parameters of the `nebular` module.

Parameter	Unit	Description
logU	–	Ionisation parameter
f_esc	–	Fraction of Lyman continuum photons escaping the galaxy
f_dust	–	Fraction of Lyman continuum photons absorbed by dust
lines_width	$km\,s^{-1}$	Line width
emission	–	Include nebular emission

Table C.18. Output parameters of the `nebular` module.

Parameter	Unit	Description
nebular.f_esc	–	Fraction of Lyman continuum photons escaping the galaxy
nebular.f_dust	–	Fraction of Lyman continuum photons absorbed by dust
dust.dust_luminosity	W	Luminosity absorbed by dust
nebular.lines_width	$km\,s^{-1}$	Line width
nebular.logU	–	Ionisation parameter

C.4. Attenuation law

C.4.1. dustatt_modified_CF00

Table C.19. Input parameters of the `dustatt_modified_CF00` module.

Parameter	Unit	Description
Av_ISM	mag	*V*-band attenuation in the interstellar medium
mu	–	Av_ISM / (Av_BC+Av_ISM)
slope_ISM	–	Power law slope of the attenuation in the ISM
slope_BC	–	Power law slope of the attenuation in the birth clouds
filters	–	Filters for which the attenuation will be computed and added to the SED information dictionary

Table C.20. Input parameters of the `dustatt_modified_CF00` module.

Parameter	Unit	Description
`attenuation.Av_ISM`	mag	*V*-band attenuation in the interstellar medium
`attenuation.Av_BC`	mag	*V*-band attenuation in the birth clouds
`attenuation.mu`	–	Av_ISM / (Av_BC+Av_ISM)
`attenuation.slope_ISM`	–	Power law slope of the attenuation in the ISM
`attenuation.slope_BC`	–	Power law slope of the attenuation in the birth clouds
`dust.luminosity`	W	Luminosity absorbed by the dust
`attenuation.[filters]`	–	Attenuation in `filters`

C.4.2. `dustatt_modified_starburst`

Table C.21. Input parameters of the `dustatt_modified_starburst` module.

Parameter	Unit	Description
`E_BV_lines`	mag	$E(B-V)$l, the colour excess of the nebular lines light for both the young and old population
`E_BV_factor`	–	Reduction factor to apply on E_BV_LINES to compute E(B-V)s the stellar continuum attenuation
`uv_bump_wavelength`	nm	Central wavelength of the UV bump
`uv_bump_width`	nm	Width (FWHM) of the UV bump
`uv_bump_amplitude`	–	Amplitude of the UV bump. For the Milky Way: 3
`powerlaw_slope`	–	Slope delta of the power law modifying the attenuation curve
`Ext_law_emission_lines`	–	Extinction law to use for attenuating the emission lines flux. Possible values are: 1, 2, 3. 1: MW, 2: LMC, 3: SMC
`Rv`	–	Ratio of total to selective extinction, A_V / E(B-V), for the extinction curve applied to emission lines. Standard value is 3.1 for MW using CCM89, but can be changed. For SMC and LMC using Pei92 the value is automatically set to 2.93 and 3.16 respectively, no matter the value you write.
`filters`	–	Filters for which the attenuation will be computed and added to the SED information dictionary

Table C.22. Output parameters of the `dustatt_modified_starburst` module.

Parameter	Unit	Description
`dust.luminosity`	W	Luminosity absorbed by the dust
`attenuation.[filters]`	–	Attenuation in `filters`
`attenuation.E_BV_lines`	mag	$E(B-V)$l, the colour excess of the nebular lines light for both the young and old population
`attenuation.E_BVs`	mag	$E(B-V)$s, the colour excess of the stellar light for both the young and old population
`attenuation.E_BV_factor`	–	Reduction factor to apply on `E_BV_lines` to compute $E(B-V)$s the stellar continuum attenuation
`attenuation.uv_bump_wavelength`	nm	Central wavelength of the UV bump
`attenuation.uv_bump_width`	nm	Width (FWHM) of the UV bump
`attenuation.uv_bump_amplitude`	–	Amplitude of the UV bump. For the Milky Way: 3
`attenuation.powerlaw_slope`	–	Slope delta of the power law modifying the attenuation curve
`attenuation.filters`	–	Filters for which the attenuation will be computed and added to the SED information dictionary

C.5. Dust emission

C.5.1. `dale2014`

Table C.23. Input parameters of the `dale2014` module.

Parameter	Unit	Description
`fracAGN`	mag	AGN fraction
`alpha`	–	Alpha slope

Table C.24. Output parameters of the `dale2014` module.

Parameter	Unit	Description
`dust.luminosity`	W	Dust luminosity (only added where there are no stellar populations)
`agn.fracAGN_dale2014`	mag	AGN fraction
`dust.alpha`	–	Alpha slope

C.5.2. `dl2007`

Table C.25. Input parameters of the `dl2007` module.

Parameter	Unit	Description
`qpah`	–	Mass fraction of PAH
`umin`	Habing	Minimum radiation field
`umax`	Habing	Maximum radiation field
`gamma`	–	Fraction illuminated from Umin to Umax

Table C.26. Output parameters of the `dl2007` module.

Parameter	Unit	Description
`dust.luminosity`	W	Dust luminosity (only added where there are no stellar populations)
`dust.qpah`	–	Mass fraction of PAH
`dust.umin`	Habing	Minimum radiation field
`dust.umax`	Habing	Maximum radiation field
`dust.gamma`	–	Fraction illuminated from Umin to Umax
`dust.mass`	kg	Dust mass

C.5.3. `dl2014` and `themis`

Table C.27. Input parameters of the `dl2014` and `themis` modules.

Parameter	Unit	Description
`qpah`	–	Mass fraction of PAH
`umin`	Habing	Minimum radiation field
`alpha`	–	Powerlaw slope $dU/dM \propto U^{\alpha}$
`gamma`	–	Fraction illuminated from Umin to Umax

Table C.28. Output parameters of the `dl2014` and `themis` modules.

Parameter	Unit	Description
`dust.luminosity`	W	Dust luminosity (only added where there are no stellar populations)
`dust.qpah`	–	Mass fraction of PAH
`dust.umin`	Habing	Minimum radiation field
`dust.alpha`	–	Powerlaw slope $dU/dM \propto U^{\alpha}$
`dust.gamma`	–	Fraction illuminated from Umin to Umax
`dust.mass`	kg	Dust mass

C.5.4. `casey2012`

Table C.29. Input parameters of the `casey2012` module.

Parameter	Unit	Description
temperature	K	Temperature of the dust
beta	–	Emissivity index of the dust
alpha	–	Mid-infrared powerlaw slope

Table C.30. Output parameters of the `casey2012` module.

Parameter	Unit	Description
dust.luminosity	W	Dust luminosity (only added where there are no stellar populations)
dust.temperature	K	Temperature of the dust
dust.beta	–	Emissivity index of the dust
dust.alpha	–	Mid-infrared powerlaw slope

C.6. Synchrotron radio emission

C.6.1. synchrotron

Table C.31. Input parameters of the `synchrotron` module.

Parameter	Unit	Description
qir	–	The value of the FIR/radio correlation coefficient
alpha	–	The slope of the power-law synchrotron emission

Table C.32. Output parameters of the `synchrotron` module.

Parameter	Unit	Description
radio.qir	–	The value of the FIR/radio correlation coefficient
radio.alpha	–	The slope of the power-law synchrotron emission

C.7. Active nucleus

C.7.1. `fritz2006`

Table C.33. Input parameters of the `fritz2006` module.

Parameter	Unit	Description
r_ratio	–	Ratio of the maximum to minimum radii of the dust torus
tau	–	Optical depth at 9.7 microns
beta	–	Beta
gamma	–	Gamma
opening_angle	degree	Full opening angle of the dust torus (Fig. 1 of Fritz et al. 2006)
psy	degree	Angle between equatorial axis and line of sight
fracAGN	–	AGN fraction

Table C.34. Output parameters of the `fritz2006` module.

Parameter	Unit	Description
agn.r_ratio	–	Ratio of the maximum to minimum radii of the dust torus
agn.tau	–	Optical depth at 9.7 microns
agn.beta	–	Beta
agn.gamma	–	Gamma
agn.opening_angle	degree	Full opening angle of the dust torus (Fig. 1 of Fritz et al. 2006)
agn.psy	degree	Angle between equatorial axis and line of sight
agn.fracAGN	–	AGN fraction

C.8. Physical property measurement

C.8.1. restframe_parameters

Table C.35. Input parameters of the restframe_parameters module.

Parameter	Unit	Description
beta_calz94	–	UV slope measured in the same way as in Calzetti et al. (1994)
D4000	–	D4000 break using the Balogh et al. (1999) definition
IRX	–	IRX computed from the GALEX FUV filter and the dust luminosity
EW_lines	–	Central wavelength of the emission lines for which to compute the equivalent width
luminosity_filters	–	Filters for which the rest-frame luminosity will be computed
colours_filters	–	Rest-frame colours to be computed

Table C.36. Output parameters of the restframe_parameters module.

Parameter	Unit	Description
param.beta_calz94	–	UV slope measured in the same way as in Calzetti et al. (1994)
param.D4000	–	D4000 break using the Balogh et al. (1999) definition
param.IRX	–	IRX computed from the GALEX FUV filter and the dust luminosity
param.EW([line])	nm	Central wavelength of the emission lines for which to compute the equivalent width
param.restframe_Lnu([filter])	$\mathrm{W\,m^{-2}}$	Filters for which the rest-frame luminosity will be computed
param.restframe_[filter1]-[filter2]	AB mag	Rest-frame colours to be computed

C.9. Intergalactic medium

C.9.1. redshifting

Table C.37. Input parameters of the redshifting module.

Parameter	Unit	Description
redshift	–	Redshift to apply to the galaxy

Table C.38. Output parameters of the redshifting module.

Parameter	Unit	Description
universe.redshift	–	Redshift to apply to the galaxy
universe.luminosity_distance	m	Luminosity distance at redshift
universe.age	Myr	Age of the universe at redshift

Appendix D: Usage of `CIGALE` executables

Table D.1. Usage of the `CIGALE` executable 1. to generate and fit models to observations (`pcigale`), 2. plot the SED, the χ^2, and the PDF (`pcigale-plots`), and 3. manage the filters in the database and plot them (`pcigale-filters`).

Executable	Command	Description
`pcigale`	`init`	Creates a skeleton configuration file
	`genconf`	Fills the configuration file according to the modules indicated
	`check`	Performs basic checks and computes the number of models
	`run`	Runs `CIGALE`
`pcigale-plots`	`sed [−nologo −type mJy, lum]`	Plots the SED in units of mJy or W
	`pdf`	Plots the PDF of the analysed physical properties
	`chi2`	Plots the χ^2 of the analysed physical properties
	`mock`	Plots the comparison between mock and true physical properties
`pcigale-filters`	`list`	Lists all the filters in the database
	`add file1,[file2, ...]`	Adds one or more filters in the database
	`del filtername,[filtername2, ...]`	Deletes one or more filters from the database
	`plot [filtername1, filtername2, ...]`	Plots one or more filters

Appendix E: Example of a `pcigale.ini` configuration file

```
# File containing the input data. The columns are 'id' (name of the
# object), 'redshift' (if 0 the distance is assumed to be 10 pc),
# 'distance' (Mpc, optional, if present it will be used in lieu of the
# distance computed from the redshift), the filter names for the fluxes,
# and the filter names with the '_err' suffix for the uncertainties. The
# fluxes and the uncertainties must be in mJy for broadband data and in
# W/m2 for emission lines. This file is optional to generate the
# configuration file, in particular for the savefluxes module.
data_file = brown14-with-dist.fits

# Optional file containing the list of physical parameters. Each column
# must be in the form module_name.parameter_name, with each line being a
# different model. The columns must be in the order the modules will be
# called. The redshift column must be the last one. Finally, if this
# parameter is not empty, cigale will not interpret the configuration
# parameters given in pcigale.ini. They will be given only for
# information. Note that this module should only be used in conjonction
# with the savefluxes module. Using it with the pdf_analysis module will
# yield incorrect results.
parameters_file =

# Avaiable modules to compute the models. The order must be kept.
# SFH:
# * sfh2exp (double exponential)
# * sfhdelayed (delayed SFH with optional exponential burst)
# * sfhdelayedbq (delayed SFH with optional constant burst/quench)
# * sfhfromfile (arbitrary SFH read from an input file)
# * sfhperiodic (periodic SFH, exponential, rectangle or delayed)
# SSP:
# * bc03 (Bruzual and Charlot 2003)
# * m2005 (Maraston 2005)
# Nebular emission:
# * nebular (continuum and line nebular emission)
# Dust attenuation:
# * dustatt_modified_CF00 (modified Charlot & Fall 2000 attenuation law)
# * dustatt_modified_starburst (modified starburst attenuaton law)
# Dust emission:
# * casey2012 (Casey 2012 dust emission models)
# * dale2014 (Dale et al. 2014 dust emission templates)
# * dl2007 (Draine & Li 2007 dust emission models)
# * dl2014 (Draine et al. 2014 update of the previous models)
# * themis (Themis dust emission models from Jones et al. 2017)
# AGN:
```

```
# * fritz2006 (AGN models from Fritz et al. 2006)
# Radio:
# * radio (synchrotron emission)
# Restframe parameters:
# * restframe_parameters (UV slope, IRX-beta, D4000, EW, etc.)
# Redshift+IGM:
# * redshifting (mandatory, also includes the IGM from Meiksin 2006)
sed_modules = sfhdelayed, bc03, nebular, dustatt_modified_starburst, dale2014,
              restframe_parameters, redshifting

# Method used for statistical analysis. Available methods: pdf_analysis,
# savefluxes.
analysis_method = pdf_analysis

# Number of CPU cores available. This computer has 96 cores.
cores = 48

# Bands to consider. To consider uncertainties too, the name of the band
# must be indicated with the _err suffix. For instance: FUV, FUV_err.
bands = galex.FUV, galex.FUV_err, galex.NUV, galex.NUV_err, sdss.up,
        sdss.up_err, sdss.gp, sdss.gp_err, sdss.rp, sdss.rp_err, sdss.ip,
        sdss.ip_err, sdss.zp, sdss.zp_err, 2mass.J, 2mass.J_err, 2mass.H,
        2mass.H_err, 2mass.Ks, 2mass.Ks_err, WISE1, WISE1_err,
        spitzer.irac.ch1, spitzer.irac.ch1_err, spitzer.irac.ch2,
        spitzer.irac.ch2_err, WISE2, WISE2_err, spitzer.irac.ch3,
        spitzer.irac.ch3_err, spitzer.irac.ch4, spitzer.irac.ch4_err, WISE3,
        WISE3_err, WISE4, WISE4_err, spitzer.mips.24, spitzer.mips.24_err

# Properties to be considered. All properties are to be given in the
# rest frame rather than the observed frame. This is the case for
# instance the equivalent widths and for luminosity densities.
properties =


# Configuration of the SED creation modules.
[sed_modules_params]

  [[sfhdelayed]]
    # e-folding time of the main stellar population model in Myr.
    tau_main = 1, 500, 1000, 2000, 3000, 4000, 5000, 6000, 7000, 8000
    # Age of the main stellar population in the galaxy in Myr. The precision
    # is 1 Myr.
    age_main = 13000
    # e-folding time of the late starburst population model in Myr.
    tau_burst = 5, 10, 25, 50, 100
    # Age of the late burst in Myr. The precision is 1 Myr.
    age_burst = 5, 10, 25, 50, 100, 200, 350, 500, 750, 1000
    # Mass fraction of the late burst population.
    f_burst = 0, 0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1, 0.25
    # Value of SFR at t = 0 in M_sun/yr.
    sfr_A = 1.0
    # Normalise the SFH to produce one solar mass.
    normalise = True

  [[bc03]]
    # Initial mass function: 0 (Salpeter) or 1 (Chabrier).
    imf = 1
    # Metalicity. Possible values are: 0.0001, 0.0004, 0.004, 0.008, 0.02,
    # 0.05.
    metallicity = 0.02
    # Age [Myr] of the separation between the young and the old star
    # populations. The default value in 10^7 years (10 Myr). Set to 0 not to
    # differentiate ages (only an old population).
    separation_age = 10

  [[nebular]]
    # Ionisation parameter
    logU = -3.0
    # Fraction of Lyman continuum photons escaping the galaxy
    f_esc = 0.0
```

```
    # Fraction of Lyman continuum photons absorbed by dust
    f_dust = 0.0
    # Line width in km/s
    lines_width = 300.0
    # Include nebular emission.
    emission = True

  [[dustatt_modified_starburst]]
    # E(B-V)l, the colour excess of the nebular lines light for both the
    # young and old population.
    E_BV_lines = 0.005, 0.01, 0.025, 0.05, 0.075, 0.10, 0.15, 0.20, 0.25, 0.30,
                 0.35, 0.40, 0.45, 0.50, 0.55, 0.60, 0.65, 0.70
    # Reduction factor to apply on E_BV_lines to compute E(B-V)s the stellar
    # continuum attenuation. Both young and old population are attenuated
    # with E(B-V)s.
    E_BV_factor = 0.25, .5, .75
    # Central wavelength of the UV bump in nm.
    uv_bump_wavelength = 217.5
    # Width (FWHM) of the UV bump in nm.
    uv_bump_width = 35.0
    # Amplitude of the UV bump. For the Milky Way: 3.
    uv_bump_amplitude = 0., 1., 2., 3.
    # Slope delta of the power law modifying the attenuation curve.
    powerlaw_slope = -.5, -.4, -.3, -.2, -.1, 0.
    # Extinction law to use for attenuating the emissio n lines flux.
    # Possible values are: 1, 2, 3. 1: MW, 2: LMC, 3: SMC. MW is modelled
    # using CCM89, SMC and LMC using Pei92.
    Ext_law_emission_lines = 1
    # Ratio of total to selective extinction, A_V / E(B-V), for the
    # extinction curve applied to emission lines.Standard value is 3.1 for
    # MW using CCM89, but can be changed.For SMC and LMC using Pei92 the
    # value is automatically set to 2.93 and 3.16 respectively, no matter
    # the value you write.
    Rv = 3.1
    # Filters for which the attenuation will be computed and added to the
    # SED information dictionary. You can give several filter names
    # separated by a & (don't use commas).
    filters = V_B90 & FUV

  [[dale2014]]
    # AGN fraction. It is not recommended to combine this AGN emission with
    # the of Fritz et al. (2006) models.
    fracAGN = 0.0
    # Alpha slope. Possible values are: 0.0625, 0.1250, 0.1875, 0.2500,
    # 0.3125, 0.3750, 0.4375, 0.5000, 0.5625, 0.6250, 0.6875, 0.7500,
    # 0.8125, 0.8750, 0.9375, 1.0000, 1.0625, 1.1250, 1.1875, 1.2500,
    # 1.3125, 1.3750, 1.4375, 1.5000, 1.5625, 1.6250, 1.6875, 1.7500,
    # 1.8125, 1.8750, 1.9375, 2.0000, 2.0625, 2.1250, 2.1875, 2.2500,
    # 2.3125, 2.3750, 2.4375, 2.5000, 2.5625, 2.6250, 2.6875, 2.7500,
    # 2.8125, 2.8750, 2.9375, 3.0000, 3.0625, 3.1250, 3.1875, 3.2500,
    # 3.3125, 3.3750, 3.4375, 3.5000, 3.5625, 3.6250, 3.6875, 3.7500,
    # 3.8125, 3.8750, 3.9375, 4.0000
    alpha = 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0

  [[restframe_parameters]]
    # UV slope measured in the same way as in Calzetti et al. (1994).
    beta_calz94 = True
    # D4000 break using the Balogh et al. (1999) definition.
    D4000 = False
    # IRX computed from the GALEX FUV filter and the dust luminosity.
    IRX = True
    # Central wavelength of the emission lines for which to compute the
    # equivalent width. The half-bandwidth must be indicated after the '/'
    # sign. For instance 656.3/1.0 means oth the nebular line and the
    # continuum are integrated over 655.3-657.3 nm.
    EW_lines =
    # Filters for which the rest-frame luminosity will be computed. You can
    # give several filter names separated by a & (don't use commas).
    luminosity_filters =
    # Rest-frame colours to be computed. You can give several colours
```

```
    # separated by a & (don't use commas).
    colours_filters =

  [[redshifting]]
    # Redshift of the objects. Leave empty to use the redshifts from the input
    # file.
    redshift = 0.


# Configuration of the statistical analysis method.
[analysis_params]
  # List of the physical properties to estimate. Leave empty to analyse
  # all the physical properties (not recommended when there are many
  # models).
  variables = stellar.m_star, dust.luminosity, attenuation.FUV, sfh.sfr100Myrs,
             param.IRX, param.beta_calz94
  # If true, save the best SED for each observation to a file.
  save_best_sed = True
  # If true, for each observation and each analysed property, save the raw
  # chi2. It occupies ~15 MB/million models/variable.
  save_chi2 = False
  # If true, for each object check whether upper limits are present and
  # analyse them.
  lim_flag = False
  # If true, for each object we create a mock object and analyse them.
  mock_flag = True
  # When redshifts are not given explicitly in the redshifting module,
  # number of decimals to round the observed redshifts to compute the grid
  # of models. To disable rounding give a negative value. Do not round if
  # you use narrow-band filters.
  redshift_decimals = 2
  # Number of blocks to compute the models and analyse the observations.
  # If there is enough memory, we strongly recommend this to be set to 1.
  blocks = 1
```