Monthly Notices
of the
ROYAL ASTRONOMICAL SOCIETY

MNRAS **452,** 1447–1467 (2015) doi:10.1093/mnras/stv1374

SLUG – stochastically lighting up galaxies – III. A suite of tools for simulated photometry, spectroscopy, and Bayesian inference with stochastic stellar populations

Mark R. Krumholz,[1]⋆ Michele Fumagalli,[2,3]⋆ Robert L. da Silva,[1] Theodore Rendahl[1] and Jonathan Parra[1]

[1] *Department of Astronomy & Astrophysics, University of California, Santa Cruz, CA 95064, USA*
[2] *Institute for Computational Cosmology and Centre for Extragalactic Astronomy, Department of Physics, Durham University, South Road, Durham DH1 3LE, UK*
[3] *Carnegie Observatories, 813 Santa Barbara Street, Pasadena, CA 91101, USA*

Accepted 2015 June 18. Received 2015 June 17; in original form 2015 February 13

ABSTRACT

Stellar population synthesis techniques for predicting the observable light emitted by a stellar population have extensive applications in numerous areas of astronomy. However, accurate predictions for small populations of young stars, such as those found in individual star clusters, star-forming dwarf galaxies, and small segments of spiral galaxies, require that the population be treated stochastically. Conversely, accurate deductions of the properties of such objects also require consideration of stochasticity. Here we describe a comprehensive suite of modular, open-source software tools for tackling these related problems. These include the following: a greatly-enhanced version of the SLUG code introduced by da Silva et al., which computes spectra and photometry for stochastically or deterministically sampled stellar populations with nearly arbitrary star formation histories, clustering properties, and initial mass functions; CLOUDY_SLUG, a tool that automatically couples SLUG-computed spectra with the CLOUDY radiative transfer code in order to predict stochastic nebular emission; BAYESPHOT, a general-purpose tool for performing Bayesian inference on the physical properties of stellar systems based on unresolved photometry; and CLUSTER_SLUG and SFR_SLUG, a pair of tools that use BAYESPHOT on a library of SLUG models to compute the mass, age, and extinction of mono-age star clusters, and the star formation rate of galaxies, respectively. The latter two tools make use of an extensive library of pre-computed stellar population models, which are included in the software. The complete package is available at http://www.slugsps.com.

Key words: methods: numerical – methods: statistical – techniques: photometric – stars: formation – galaxies: star clusters: general – galaxies: stellar content.

1 INTRODUCTION

Stellar population synthesis (SPS) is a critical tool that allows us to link the observed light we receive from unresolved stellar populations to the physical properties (e.g. mass, age) of the emitting stars. Reflecting this importance, over the years numerous research groups have written and distributed SPS codes such as STARBURST99 (Leitherer et al. 1999; Vázquez & Leitherer 2005), FSPS (Conroy, Gunn & White 2009; Conroy & Gunn 2010), PEGASE (Fioc & Rocca-Volmerange 1997), and GALAXEV (Bruzual & Charlot 2003). All these codes perform essentially the same computation. One adopts a star formation history (SFH) and an initial mass function (IMF) to determine the present-day distribution of stellar masses and ages. Next, using a set of stellar evolutionary tracks and atmospheres that give the luminosity (either frequency-dependent or integrated over some filter) for a star of a particular mass and age, one integrates the stellar luminosity weighted by the age and mass distributions. These codes differ in the range of functional forms they allow for the IMF and SFH, and the evolutionary tracks and model atmospheres they use, but the underlying computation is much the same in all of them. While this approach is adequate for many applications, it fails for systems with low masses and star formation rates (SFRs) because it implicitly assumes that the stellar mass and age distributions are well sampled. This is a very poor assumption both in star-forming dwarf galaxies and in resolved segments of larger galaxies.

Significantly less work has been done in extending SPS methods to the regime where the IMF and SFH are not well sampled. There

⋆ E-mail: mkrumhol@ucsc.edu (MRK); fumagalli.astro@gmail.com (MF) are a number of codes available for simulating a simple stellar population (SSP; i.e. one where all the stars are the same age, so the SFH is described by a δ distribution) where the IMF is not well sampled (Maíz Apellániz 2009; Popescu & Hanson 2009, 2010b,a; Fouesneau & Lançon 2010; Fouesneau et al. 2012, 2014; Anders et al. 2013; de Meulenaer et al. 2013, 2014, 2015), and a great deal of analytic work has also been performed on this topic (Cerviño & Valls-Gabaud 2003; Cerviño & Luridiana 2004, 2006) – see Cerviño (2013) for a recent review. However, these codes only address the problem of stochastic sampling of the IMF; for non-SSPs, stochastic sampling of the SFH proves to be a more important effect (Fumagalli, da Silva & Krumholz 2011; da Silva, Fumagalli & Krumholz 2014).

To handle this problem, we introduced the stochastic SPS code SLUG (da Silva, Fumagalli & Krumholz 2012), which includes full stochasticity in both the IMF and the SFH. Crucially, SLUG properly handles the clustered nature of star formation (e.g. Lada & Lada 2003; Krumholz 2014). This has two effects. First, clustering itself can interact with IMF sampling so that the total mass distribution produced by a clustered population is not identical to that of a non-clustered population drawn from the same underlying IMF. Secondly and perhaps more importantly, clustering causes large short-time-scale fluctuations in the SFR even in galaxies whose mean SFR averaged over longer time-scales is constant. Since its introduction, this code has been used in a number of applications, including explaining the observed deficit of H α emission relative to far-ultraviolet (FUV) emission in dwarf galaxies (Fumagalli et al. 2011; Andrews et al. 2013, 2014), quantifying the stochastic uncertainties in SFR indicators (da Silva et al. 2014), analysing the properties of star clusters in dwarf galaxies in the ANGST survey (Cook et al. 2012), and analysing Lyman continuum radiation from high-redshift dwarf galaxies (Forero-Romero & Dijkstra 2013). The need for a code with stochastic capabilities is likely to increase in the future, as studies of local galaxies such as PHAT (Dalcanton et al. 2012), HERACLES (Leroy et al. 2013), and LEGUS (Calzetti et al. 2015), and even studies in the high-redshift Universe (e.g. Jones et al. 2013a,b) increasingly push to smaller spatial scales and lower rates of star formation, where stochastic effects become increasingly important.

In this paper we describe a major upgrade and expansion of SLUG, intended to make it a general-purpose solution for the analysis of stochastic stellar populations. This new version of SLUG allows essentially arbitrary functional forms for both the IMF and the SFH, allows a wide range of stellar evolutionary tracks and atmosphere models, and can output both spectroscopy and photometry in a list of >100 filters. It can also include the effects of reprocessing of the light by interstellar gas and by stochastically-varying amounts of dust, and can interface with the CLOUDY photoionization code (Ferland et al. 2013) to produce predictions for stochastically-varying nebular emission. Finally, we have coupled SLUG to a new set of tools for solving the 'inverse problem' in stochastic SPS: given a set of observed photometric properties, infer the posterior probability distribution for the properties of the underlying stellar population, in the case where the mapping between such properties and the photometry is stochastic and therefore non-deterministic (e.g. da Silva et al. 2014). The full software suite is released under the GNU Public License, and is freely available from http://www.slugsps.com.[1]

In the remainder of this paper, we describe SLUG and its companion software tools in detail (Section 2), and then provide a series of demonstrations of the capabilities of the upgraded version of the code (Section 3). We end with a summary and discussion of future prospects for this work (Section 4).

2 THE SLUG SOFTWARE SUITE

The SLUG software suite is a collection of tools designed to solve two related problems. The first is to determine the probability distribution function (PDF) of observable quantities (spectra, photometry, etc.) that are produced by a stellar population characterized by a specified set of physical parameters (IMF, SFH, cluster mass function, and an array of additional ancillary inputs). This problem is addressed by the core SLUG code (Section 2.1) and its adjunct CLOUDY_SLUG (Section 2.2), which perform Monte Carlo simulations to calculate the distribution of observables. The second problem is to use those PDFs to solve the inverse problem: given a set of observed properties, what should we infer about the physical properties of the stellar population producing those observables? The BAYESPHOT package provides a general solution to this problem (Section 2.3), and the SFR_SLUG and CLUSTER_SLUG packages specialize this general solution to the problem of inferring SFRs for continuously star-forming galaxies, and masses, ages, and extinctions from SSPs, respectively (Section 2.4). The entire software suite is available from http://www.slugsps.com, and extensive documentation is available at http://slug2.readthedocs.org/en/latest/.

2.1 SLUG: a highly flexible tool for stochastic SPS

The core of the software suite is the stochastic SPS code SLUG. The original SLUG code is described in da Silva et al. (2012), but the code described here is a complete re-implementation with greatly expanded capabilities. The code operates in two main steps; first it generates the spectra from the stellar population (Section 2.1.1) and then it post-processes the spectra to include the effects of nebular emission and dust absorption, and to produce photometric predictions as well as spectra (Section 2.1.2). In this section we limit ourselves to a top-level description of the physical model, and provide some more details on the numerical implementation in Appendices A, B, and C.

2.1.1 Generating the stellar population

SLUG can simulate both SSPs (i.e. ones where all the stars are coeval) and composite ones. We begin by describing the SSP model, since composite stellar populations in SLUG are built from collections of SSPs. For an SSP, the stellar population is described by an age, a total mass, and an IMF. SLUG allows nearly-arbitrary functional forms for the IMF, and is not restricted to a set of predetermined choices; see Appendix A1 for details.

In non-stochastic SPS codes, once an age, mass, and IMF are chosen, the standard procedure is to use a set of stellar tracks and atmospheres to predict the luminosity (either frequency dependent, or integrated over a specified filter) as a function of stellar mass and age, and to integrate the mass-dependent luminosity multiplied by the stellar mass distribution to compute the output spectrum at a specified age. SLUG adds an extra step: instead of evaluating an

[1] As of this writing http://www.slugsps.com is hosted on Google Sites, and thus is inaccessible from mainland China. Chinese users can access the source code from https://bitbucket.org/krumholz/slug2, and should contact the authors by email for the ancillary data products.

integral, it directly draws stars from the IMF until the desired total mass has been reached. As emphasized by a number of previous authors (e.g. Weidner & Kroupa 2006; Haas & Anders 2010; Cerviño 2013; Popescu & Hanson 2014), when drawing a target mass $M_{\rm target}$ rather than a specified number of objects from a PDF, one must also choose a sampling method to handle the fact that, in general, it will not be possible to hit the target mass exactly. Many sampling procedures have been explored in the literature, and SLUG provides a large number of options, as described in Appendix A1.

Once a set of stars is chosen, SLUG proceeds much like a conventional SPS code. It uses a chosen set of tracks and atmospheres to determine the luminosity of each star, either wavelength dependent or integrated over one or more photometric filters, at the specified age. It then sums over all stars to determine the integrated light output. Details of the available sets of tracks and atmospheres, and SLUG's method for interpolating them, are provided in Appendix A2.

Composite stellar populations in SLUG consist of a collection of 'star clusters', each consisting of an SSP, plus a collection of 'field stars' whose ages are distributed continuously. In practice, this population is described by four parameters beyond those that describe SSPs: the fraction of stars formed in clusters as opposed to the field $f_{\rm c}$, the cluster mass function (CMF), cluster lifetime function (CLF), and SFH. As with the IMF, the latter three are distributions, which can be specified using nearly arbitrary functional forms and a wide range of sampling methods as described in Appendix A1. For a simulation with a composite stellar population, SLUG performs a series of steps: (1) at each user-requested output time,[2] SLUG uses the SFH and the cluster fraction $f_{\rm c}$ to compute the additional stellar mass expected to have formed in clusters and out of clusters (in 'the field') since the previous output; (2) for the non-clustered portion of the star formation, SLUG draws the appropriate mass in individual stars from the IMF, while for the clustered portion it draws a cluster mass from the CMF, and then it fills each cluster with stars drawn from the IMF; (3) each star and star cluster formed is assigned an age between the current output and the previous one, selected to produce a realization of the input SFH;[3] (4) each cluster that is formed is assigned a lifetime drawn from the CLF, which is the time at which the cluster is considered dispersed.

The end result of this procedure is that, at each output time, SLUG has constructed a 'galaxy' consisting of a set of star clusters and field stars, each with a known age. Once the stellar population has been assembled, the procedure for determining spectra and photometry is simply a sum over the various individual simple populations. In addition to computing the integrated spectrum of the entire population, SLUG can also report individual spectra for each cluster that has not disrupted (i.e. where the cluster age is less than the value drawn from the CLF for that cluster). Thus the primary output consists of an integrated monochromatic luminosity L_λ for the entire stellar population, and a value $L_{\lambda,\,i}$ for the ith remaining cluster, at each output time.

Finally, we note that SLUG is also capable of skipping the sampling procedure and evaluating the light output of stellar populations by integrating over the IMF and SFH, exactly as in a conventional, non-stochastic SPS code. In this case, SLUG essentially emulates STARBURST99 (Leitherer et al. 1999; Vázquez & Leitherer 2005), except for subtle differences in the interpolation and numerical integration schemes used (see Appendix A2). SLUG can also behave 'semi-stochastically', evaluating stars' contribution to the light output using integrals to handle the IMF up to some mass, and using stochastic sampling at higher masses.

2.1.2 *Post-processing the spectra*

Once SLUG has computed a spectrum for a simple or composite stellar population, it can perform three additional post-processing steps. First, it can provide an approximate spectrum after the starlight has been processed by the surrounding H II region. In this case, the nebular spectrum is computed for an isothermal, constant-density H II region, within which it is assumed that He is all singly ionized. Under these circumstances, the photoionized volume V, electron density $n_{\rm e}$, and hydrogen density $n_{\rm H}$ obey the relation

$$\phi Q(\mathrm{H}^0) = \alpha^{(\mathrm{B})}(T) n_{\rm e} n_{\rm H} V, \tag{1}$$

where

$$Q(\mathrm{H}^0) = \int_{h\nu=I(\mathrm{H}^0)}^{\infty} \frac{L_\nu}{h\nu}\,\mathrm{d}\nu \tag{2}$$

is the hydrogen-ionizing luminosity, $L_\nu = \lambda^2 L_\lambda / c$ is the luminosity per unit frequency, $I(\mathrm{H}^0) = 13.6$ eV is the ionization potential of neutral hydrogen, $n_{\rm e} = (1 + x_{\rm He}) n_{\rm H}$, $x_{\rm He}$ is the He abundance relative to hydrogen (assumed to be 0.1), ϕ is the fraction of H-ionizing photons that are absorbed by hydrogen atoms within the H II region rather than escaping or being absorbed by dust, and $\alpha^{(\mathrm{B})}(T)$ is the temperature-dependent case B recombination coefficient. The nebular luminosity can then be written as

$$L_{\lambda,\mathrm{neb}} = \gamma_{\rm neb} n_{\rm e} n_{\rm H} V = \gamma_{\rm neb} \phi \frac{Q(\mathrm{H}^0)}{\alpha^{(\mathrm{B})}(T)}, \tag{3}$$

where $\gamma_{\rm neb}$ is the wavelength-dependent nebular emission coefficient. Our calculation of this quantity includes the following processes: free–free and bound–free emission arising from electrons interacting with H^+ and He^+, two-photon emission from neutral hydrogen in the 2s state, H recombination lines, and non-H lines computed approximately from tabulated values. Full details on the method by which we perform this calculation are given in Appendix B. The composite spectrum after nebular processing is zero at wavelength shorter than 912 Å (corresponding to the ionization potential of hydrogen), and the sum of intrinsic stellar spectrum L_λ and the nebular spectrum $L_{\lambda,\mathrm{neb}}$ at longer wavelengths. SLUG reports this quantity both cluster-by-cluster and for the galaxy as a whole.

Note that the treatment of nebular emission included in SLUG is optimized for speed rather than accuracy, and will produce significantly less accurate results than CLOUDY_SLUG (see Section 2.2). The major limitations of the built-in approach are as follows: (1) because SLUG's built-in calculation relies on pre-tabulated values for the metal line emission and assumes either a constant or a similarly pre-tabulated temperature (which significantly affects the continuum emission), it misses the variation in emission caused by the

[2] SLUG can either output results at specified times, or the user can specify a distribution from which the output time is to be drawn. The latter capability is useful when we wish to sample a distribution of times continuously so that the resulting data set can be used to infer ages from observations – see Section 2.4.

[3] One subtlety to note here is that the choice of output grid can produce non-trivial modifications of the statistical properties of the output in cases where the expected mass of stars formed is much smaller than the maximum possible cluster mass. For example, consider a simulation with a constant SFR and an output grid chosen such that the expected mass of stars formed per output time is 10^4 M$_\odot$. If the sampling method chosen is `STOP_NEAREST` (see Appendix A1), the number of 10^6 M$_\odot$ clusters formed will typically be smaller than if the same SFH were used but the outputs were spaced 100 times further apart, giving an expected mass of 10^6 M$_\odot$ between outputs.

fact that H II regions exist at a range of densities and ionization parameters, which in turn induces substantial variations in their nebular emission (e.g. Verdolini et al. 2013; Yeh et al. 2013); (2) SLUG's built-in calculation assumes a uniform density H II region, an assumption that will fail at high ionizing luminosities and densities due to the effects of radiation pressure (Dopita et al. 2002; Krumholz, McKee & Tumlinson 2009; Draine 2011; Yeh & Matzner 2012; Yeh et al. 2013); (3) SLUG's calculation correctly captures the effects of stochastic variation in the total ionizing luminosity, but it does not capture the effects of variation in the spectral shape of the ionizing continuum, which preliminary testing suggests can cause variations in line luminosities at the few tenths of a dex level even for fixed total ionizing flux. The reason for accepting these limitations is that the assumptions that cause them also make it possible to express the nebular emission in terms of a few simple parameter that can be pre-tabulated, reducing the computational cost of evaluating the nebular emission by orders of magnitude compared to a fully accurate calculation with CLOUDY_SLUG.

The second post-processing step available is that SLUG can apply extinction in order to report an extincted spectrum, both for the pure stellar spectrum and for the nebula-processed spectrum. Extinction can be either fixed to a constant value for an entire simulation, or it can be drawn from a specified PDF. In the latter case, for simulations of composite stellar populations, every cluster will have a different extinction. More details are given in Appendix B.

As a third and final post-processing step, SLUG can convolve all the spectra it computes with one or more filter response functions in order to predict photometry. SLUG includes the large list of filter response functions maintained as part of FSPS (Conroy & Gunn 2010; Conroy, White & Gunn 2010), as well as a number of *Hubble Space Telescope* (*HST*) filters[4] not included in FSPS; at present, more than 130 filters are available. As part of this calculation, SLUG can also output the bolometric luminosity, and the photon luminosity in the H^0, He^0, and He^+ ionizing continua.

2.2 CLOUDY_SLUG: stochastic nebular line emission

In broad outlines, the CLOUDY_SLUG package is a software interface that automatically takes spectra generated by SLUG and uses them as inputs to the CLOUDY radiative transfer code (Ferland et al. 2013). CLOUDY_SLUG then extracts the continuous spectra and lines returned by CLOUDY, convolves them with the same set of filters as in the original SLUG calculation in order to predict photometry. The user can optionally also examine all of CLOUDY's detailed outputs. As with the core SLUG code, details on the software implementation are given in Appendix C.

When dealing with composite stellar populations, calculating the nebular emission produced by a stellar population requires making some physical assumptions about how the stars are arranged, and CLOUDY_SLUG allows two extreme choices that should bracket reality. One extreme is to assume that all stars are concentrated in a single point of ionizing radiation at the centre of a single H II region. We refer to this as integrated mode, and in this mode the only free parameters to be specified by a user are the chemical composition of the gas into which the radiation propagates and its starting density.

The opposite assumption, which we refer to as cluster mode, is that every star cluster is surrounded by its own H II region, and that the properties of these regions are to be computed individually using the stellar spectrum of each driving cluster and only then summed to produce a composite output spectrum. At present cluster mode does not consider contributions to the nebular emission from field stars, and so should not be used when the cluster fraction $f_c < 1$. In the cluster case, the H II regions need not all have the same density or radius; indeed, one would expect a range of both properties to be present, since not all star clusters have the same age or ionizing luminosity. We handle this case using a simplified version of the H II population synthesis model introduced by Verdolini et al. (2013) and Yeh et al. (2013). The free parameters to be specified in this model are the chemical composition and the density in the ambient neutral ISM around each cluster. For an initially neutral region of uniform hydrogen number density n_H, the radius of the H II at a time t after it begins expanding is well approximated by (Krumholz & Matzner 2009)

$$r_{II} = r_{ch}\left(x_{rad}^{7/2} + x_{gas}^{7/2}\right)^{2/7} \tag{4}$$

$$x_{rad} = \left(2\tau^2\right)^{1/4} \tag{5}$$

$$x_{gas} = \left(\frac{49}{36}\tau^2\right)^{2/7} \tag{6}$$

$$\tau = \frac{t}{t_{ch}} \tag{7}$$

$$r_{ch} = \frac{\alpha_4^{(B)}}{12\pi\phi_{dust}}\left(\frac{\epsilon_0}{2.2k_B T_{II}}\right)^2 f_{trap}^2 \frac{\psi^2 Q(H^0)}{c^2} \tag{8}$$

$$t_{ch} = \left(\frac{4\pi\mu m_H n_H c r_{ch}^4}{3 f_{trap} Q(H^0)\psi I(H^0)}\right)^{1/2}, \tag{9}$$

where $\alpha_4^{(B)} = 2.59 \times 10^{-13}$ cm^3 s^{-1} is the case B recombination coefficient at 10^4 K, $T_{II} = 10^4$ is the typical H II region temperature, $f_{trap} = 2$ is a trapping factor that accounts for stellar wind and trapped infrared radiation pressure, $\psi = 3.2$ is the mean photon energy in Rydberg for a fully sampled IMF at zero age,[5] and $\mu = 1.33$ is the mean mass per hydrogen nucleus for gas with the usual helium abundance $x_{He} = 0.1$. The solution includes the effects of both radiation and gas pressure in driving the expansion. Once the radius is known, the density near the H II region centre (the parameter required by CLOUDY) can be computed from the usual ionization balance argument,

$$n_{II} = \left(\frac{3Q(H^0)}{4.4\pi\alpha_4^{(B)} r_{II}^3}\right)^{1/2}. \tag{10}$$

Note that the factor of 4.4 in the denominator accounts for the extra free electrons provided by helium, assuming it is singly ionized. Also note that this will not give the correct density during the brief radiation pressure-dominated phase early on in the evolution, but that this period is very brief (though the effects of the extra boost provided by radiation pressure can last much longer), and no simple

[4] These filters were downloaded from http://www.stsci.edu/~WFC3/UVIS/SystemThroughput/ and http://www.stsci.edu/hst/acs/analysis/throughputs for the UVIS and ACS instruments, respectively.

[5] This value of ψ is taken from Murray & Rahman (2010), and is based on a Chabrier (2005) IMF and a compilation of empirically measured ionizing luminosities for young stars. However, alternative IMFs and methods of computing the ionizing luminosity give results that agree to tens of percent.

analytic approximation for this density is available (Krumholz & Matzner 2009). Also note that, although CLOUDY_SLUG presently only supports this parametrization of H II region radius and density versus time, the code is simply a PYTHON script. It would therefore be extremely straightforward for a user who prefers a different parametrization to alter the script to supply it.

In cluster mode, CLOUDY_SLUG uses the approximation described above to compute the density of the ionized gas in the vicinity of each star cluster, which is then passed as an input to CLOUDY along with the star cluster's spectrum. Note that computations in cluster mode are much more expensive than those in integrated mode, since the latter requires only a single call to CLOUDY per timestep, while the former requires one per cluster. To ease the computational requirements slightly, in cluster mode one can set a threshold ionizing luminosity below which the contribution to the total nebular spectrum is assumed to be negligible, and is not computed.

2.3 BAYESPHOT: Bayesian inference from stellar photometry

2.3.1 Description of the method

Simulating stochastic stellar populations is useful, but to fully exploit this capability we must tackle the inverse problem: given an observed set of stellar photometry, what should we infer about the physical properties of the underlying stellar population in the regime where the mapping between physical and photometric properties is non-deterministic? A number of authors have presented numerical methods to tackle problems of this sort, mostly in the context of determining the properties of star clusters with stochastically sampled IMFs (Popescu & Hanson 2009, 2010a,b; Fouesneau & Lançon 2010; Asa'd & Hanson 2012; Fouesneau et al. 2012; Popescu, Hanson & Elmegreen 2012; Anders et al. 2013; de Meulenaer et al. 2013, 2014, 2015), and in some cases in the context of determining SFRs from photometry (da Silva et al. 2014). Our method here is a generalization of that developed in da Silva et al. (2014), and has a number of advantages as compared to earlier methods, both in terms of its computational practicality and its generality.

Consider stellar systems characterized by a set of N physical parameters $\boldsymbol{x} = (x_1, x_2, \dots, x_N)$; in the example of star clusters below we will have $N = 3$, with x_1, x_2, and x_3 representing the logarithm of the mass, the logarithm of the age, and the extinction, while for galaxies forming stars at a continuous rate we will have $N = 1$, with x_1 representing the logarithm of the SFR. The light output of these systems is known in M photometric bands; let $\boldsymbol{y} = (y_1, y_2, \dots, y_M)$ be a set of photometric values, for example magnitudes in some set of filters. Suppose that we observe a stellar population in these bands and measure a set of photometric values $\boldsymbol{y}_{\rm obs}$, with some errors $\boldsymbol{\sigma}_y = (\sigma_{y_1}, \sigma_{y_2}, \dots, \sigma_{y_M})$, which for simplicity we will assume are Gaussian. We wish to infer the posterior probability distribution for the physical parameters given the observed photometry and photometric errors, $p(\boldsymbol{x} \mid \boldsymbol{y}_{\rm obs}; \boldsymbol{\sigma}_y)$.

Following da Silva et al. (2014), we compute the posterior probability via implied conditional regression coupled to kernel density estimation. Let the joint PDF of physical and photometric values for the population of all the stellar populations under consideration be $p(\boldsymbol{x}, \boldsymbol{y})$. We can write the posterior probability distribution we seek as

$$p(\boldsymbol{x} \mid \boldsymbol{y}) \propto \frac{p(\boldsymbol{x}, \boldsymbol{y})}{p(\boldsymbol{y})}, \tag{11}$$

where $p(\boldsymbol{y})$ is the distribution of the photometric variables alone, i.e.

$$p(\boldsymbol{y}) \propto \int p(\boldsymbol{x}, \boldsymbol{y}) \, \mathrm{d}\boldsymbol{x}. \tag{12}$$

If we have an exact set of photometric measurements $\boldsymbol{y}_{\rm obs}$, with no errors, then the denominator in equation (11) is simply a constant that will normalize out, and the posterior probability distribution we seek is distributed simply as

$$p(\boldsymbol{x} \mid \boldsymbol{y}_{\rm obs}) \propto p(\boldsymbol{x}, \boldsymbol{y}_{\rm obs}). \tag{13}$$

In this case, the problem of computing $p(\boldsymbol{x} \mid \boldsymbol{y}_{\rm obs})$ reduces to that of computing the joint physical–photometric PDF $p(\boldsymbol{x}, \boldsymbol{y})$ at any given set of observed photometric values $\boldsymbol{y}_{\rm obs}$.

For the more general case where we do have errors, we first note that the posterior probability distribution for the true photometric value $\boldsymbol{y}$ is given by the prior probability distribution for photometric values multiplied by the likelihood function associated with our measurements. The prior PDF of photometric values is simply $p(\boldsymbol{y})$ as given by equation (12), and for a central observed value of $\boldsymbol{y}_{\rm obs}$ with errors $\boldsymbol{\sigma}_y$, the likelihood function is simply a Gaussian. Thus the PDF of $\boldsymbol{y}$ given our observations is

$$p(\boldsymbol{y} \mid \boldsymbol{y}_{\rm obs}) \propto p(\boldsymbol{y}) G(\boldsymbol{y} - \boldsymbol{y}_{\rm obs}, \boldsymbol{\sigma}_y), \tag{14}$$

where

$$G(\boldsymbol{y}, \boldsymbol{\sigma}_y) \propto \exp\left[-\left(\frac{y_1^2}{2\sigma_{y_1}^2} + \frac{y_2^2}{2\sigma_{y_2}^2} + \dots + \frac{y_M^2}{2\sigma_{y_M}^2}\right)\right] \tag{15}$$

is the usual multidimensional Gaussian function. The posterior probability for the physical parameters is then simply the convolution of equations (11) and (14), i.e.

$$\begin{aligned} p(\boldsymbol{x} \mid \boldsymbol{y}_{\rm obs}) &\propto \int p(\boldsymbol{x} \mid \boldsymbol{y}) \, p(\boldsymbol{y} \mid \boldsymbol{y}_{\rm obs}) \, \mathrm{d}\boldsymbol{y} \\ &\propto \int p(\boldsymbol{x}, \boldsymbol{y}) \, G(\boldsymbol{y} - \boldsymbol{y}_{\rm obs}, \boldsymbol{\sigma}_y) \, \mathrm{d}\boldsymbol{y}. \end{aligned} \tag{16}$$

Note that we recover the case without errors, equation (13), in the limit where $\boldsymbol{\sigma}_y \to 0$, because in that case the Gaussian $G(\boldsymbol{y} - \boldsymbol{y}_{\rm obs}, \boldsymbol{\sigma}_y) \to \delta(\boldsymbol{y} - \boldsymbol{y}_{\rm obs})$.

We have therefore reduced the problem of computing $p(\boldsymbol{x} \mid \boldsymbol{y}_{\rm obs})$ to that of computing $p(\boldsymbol{x}, \boldsymbol{y})$, the joint PDF of physical and photometric parameters, for our set of stellar populations. To perform this calculation, we use SLUG to create a large library of models for the type of stellar population in question. We draw the physical properties of the stellar populations (e.g. star cluster mass, age, and extinction) in our library from a distribution $p_{\rm lib}(\boldsymbol{x})$, and for each stellar population in the library we have a set of physical and photometric parameters $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$. We estimate $p(\boldsymbol{x}, \boldsymbol{y})$ using a kernel density estimation technique. Specifically, we approximate this PDF as

$$p(\boldsymbol{x}, \boldsymbol{y}) \propto \sum_i w_i K(\boldsymbol{x} - \boldsymbol{x}_i, \boldsymbol{y} - \boldsymbol{y}_i; \boldsymbol{h}), \tag{17}$$

where w_i is the weight we assign to sample i, $K(\boldsymbol{x}; \boldsymbol{h})$ is our kernel function, $\boldsymbol{h} = (h_{x_1}, h_{x_2}, \dots, h_{x_N}, h_{y_1}, h_{y_2}, \dots, h_{y_M})$ is our bandwidth parameter (see below), and the sum runs over every simulation in our library. We assign weights to ensure that the distribution of physical parameters $\boldsymbol{x}$ matches whatever prior probability distributions we wish to assign for them. If we let $p_{\rm prior}(\boldsymbol{x})$ represent our priors, then this is

$$w_i = \frac{p_{\rm prior}(\boldsymbol{x}_i)}{p_{\rm lib}(\boldsymbol{x}_i)}. \tag{18}$$

Note that, although we are free to choose $p_{\rm lib}(\boldsymbol{x}) = p_{\rm prior}(\boldsymbol{x})$ and thus weight all samples equally, it is often advantageous for numerical or astrophysical reasons not to do so, because then we can distribute our sample points in a way that is chosen to maximize our knowledge of the shape of $p(\boldsymbol{x}, \boldsymbol{y})$ with the fewest possible realizations. As a practical example, we know that photometric outputs will fluctuate more for galaxies with low SFRs than for high ones, so the library we use for SFR_SLUG (see below) is weighted to contain more realizations at low than high SFR.

We choose to use a Gaussian kernel function, $K(\boldsymbol{x}; \boldsymbol{h}) = G(\boldsymbol{x}, \boldsymbol{h})$, because this presents significant computational advantages. Specifically, with this choice, equation (16) becomes

$$p(\boldsymbol{x} \mid \boldsymbol{y}_{\rm obs}) \propto \sum_i w_i G(\boldsymbol{x} - \boldsymbol{x}_i, \boldsymbol{h}_x) \int G(\boldsymbol{y} - \boldsymbol{y}_{\rm obs}, \boldsymbol{\sigma}_y)\, G(\boldsymbol{y} - \boldsymbol{y}_i, \boldsymbol{h}_y)\, {\rm d}\boldsymbol{y} \tag{19}$$

$$\propto \sum_i w_i G\left(\boldsymbol{x} - \boldsymbol{x}_i, \boldsymbol{h}_x\right) G(\boldsymbol{y}_{\rm obs} - \boldsymbol{y}_i, \sqrt{\boldsymbol{\sigma}_y^2 + \boldsymbol{h}_y^2}) \tag{20}$$

$$\equiv \sum_i w_i G((\boldsymbol{x} - \boldsymbol{x}_i, \boldsymbol{y}_{\rm obs} - \boldsymbol{y}_i), \boldsymbol{h}'), \tag{21}$$

where $\boldsymbol{h}_x = (h_{x_1}, h_{x_2}, \ldots, h_{x_N})$ is the bandwidth in the physical dimensions, $\boldsymbol{h}_y = (h_{y_1}, h_{y_2}, \ldots, h_{y_M})$ is the bandwidth in the photometric dimensions, and the quadrature sum $\sqrt{\boldsymbol{h}_y^2 + \boldsymbol{\sigma}_y^2}$ is understood to be computed independently over every index in $\boldsymbol{\sigma}_y$ and $\boldsymbol{h}_y$. The new quantity we have introduced,

$$\boldsymbol{h}' = \left(\boldsymbol{h}_x, \sqrt{\boldsymbol{h}_y^2 + \boldsymbol{\sigma}_y^2}\right), \tag{22}$$

is simply a modified bandwidth in which the bandwidth in the photometric dimensions has been broadened by adding the photometric errors in quadrature sum with the bandwidths in the corresponding dimensions. Note that, in going from equation (19) to (20) we have invoked the result that the convolution of two Gaussians is another Gaussian, whose width is the quadrature sum of the widths of the two input Gaussians, and whose centre is located at the difference between the two input centres.

As an adjunct to this result, we can also immediately write down the marginal probabilities for each of the physical parameters in precisely the same form. The marginal posterior probability distribution for x_1 is simply

$$p(x_1 \mid \boldsymbol{y}_{\rm obs}) \propto \int p(\boldsymbol{x} \mid \boldsymbol{y}_{\rm obs})\, {\rm d}x_2\, {\rm d}x_3 \ldots {\rm d}x_N \tag{23}$$

$$\propto \sum_i w_i G(x_1 - x_{1,i}, h_1)\, G\left(\boldsymbol{y}_{\rm obs} - \boldsymbol{y}_i, \sqrt{\boldsymbol{\sigma}_y^2 + \boldsymbol{h}_y^2}\right), \tag{24}$$

and similarly for all other physical variables. This expression also immediately generalizes to the case of joint marginal probabilities of the physical variables. We have therefore succeeded in writing down the posterior probability distribution for all the physical properties, and the marginal posterior distributions of any of them individually or in combination, via a kernel density estimation identical to that given by equation (17). One advantage of equations (21) and (24) is that they are easy to evaluate quickly using standard numerical methods. Details on our software implementation are given in Appendix C.

2.3.2 Error estimation and properties of the library

An important question for BAYESPHOT, or any similar method, is how large a library must be in order to yield reliable results. Of course this depends on what quantity is being estimated and on the shape of the underlying distribution. Full, rigorous, error analysis is best accomplished by bootstrap resampling. However, we can, without resorting to such a computationally intensive procedure, provide a useful rule of thumb for how large a library must be so that the measurements rather than the library are the dominant source of uncertainty in the resulting derived properties.

Consider a library of n simulations, from which we wish to measure the value X_q that delineates a certain quantile (i.e. the percentile $p = 100q$) of the underlying distribution from which the samples are drawn. Let x_i for $i = 1, \ldots, n$ be the samples in our library, ordered from smallest to largest. Our central estimate for the value of X_q that delineates quintile q, will naturally be x_{qn}. (Throughout this argument we will assume that n is large enough that we need not worry about the fact that ranks in the list, such as qn, will not exactly be integers. Extending this argument to include the necessary continuity correction adds mathematical complication but does not change the basic result.) For example, we may have a library of 10^6 simulations of galaxies at a particular SFR (or in a particular interval of SFRs), and wish to know what ionizing luminosity corresponds to the 95th percentile at that SFR. Our estimate for the 95th percentile ionizing luminosity will simply be the ionizing luminosity of the 950 000th simulation in the library.

To place confidence intervals around this estimate, note that the probability that a randomly chosen member of the population will have a value $x < x_q$ is q, and conversely the probability that $x > x_q$ is $1 - q$. Thus in our library of n simulations, the probability that we have exactly k samples for which $x < x_q$ is given by the binomial distribution,

$${\rm Pr}(k) = \frac{n!}{(n-k)!} q^k (1-q)^{n-k}. \tag{25}$$

When $n \gg 1$, $k \gg 1$, and $n - k \gg 1$, the binomial distribution approaches a Gaussian distribution of central value qn and standard deviation $\sqrt{q(1-q)n}$:

$${\rm Pr}(k) \approx \frac{1}{\sqrt{2\pi q(1-q)n}} \exp\left[-\frac{(k-qn)^2}{2q(1-q)n}\right]. \tag{26}$$

That is, the position n in our library of simulations such that $x_i < x_q$ for all $i < n$, and $x_i > x_q$ for all $i > n$, is distributed approximately as a Gaussian, centred at qn, with dispersion $\sqrt{q(1-q)n}$. This makes it easy to compute confidence intervals on x_q, because it reduces the problem that of computing confidence intervals on a Gaussian. Specifically, if we wish to compute the central value and confidence interval c (e.g. $c = 0.68$ is the 68 per cent confidence interval) for the rank r corresponding to percentile q, this is

$$r \approx qn \pm \sqrt{2q(1-q)n}\, {\rm erf}^{-1}(c). \tag{27}$$

The corresponding central value for X_q (as opposed to its rank in the list) is x_{qn}, and the confidence interval is

$$\left(x_{qn - \sqrt{2q(1-q)n}\,{\rm erf}^{-1}(c)}, x_{qn + \sqrt{2q(1-q)n}\,{\rm erf}^{-1}(c)}\right). \tag{28}$$

In our example of determining the 95th percentile from a library of 10^6 simulations, our central estimate of this value is the $x_{950\,000}$, the

950 000th simulation in the library, and the 90 per cent confidence interval is 359 ranks on either side of this. That is, our 90 per cent confidence interval extends from $x_{949\,641}$ to $x_{950\,359}$.

This method may be used to define confidence intervals on any quantile derived from a library of SLUG simulations. Of course this does not specifically give confidence intervals on the results derived when such a library is used as the basis of a Bayesian inference from BAYESPHOT. However, this analysis can still provide a useful constraint: the error on the values derived from BAYESPHOT cannot be less than the errors on the underlying photometric distribution we have derived from the library. Thus if we have a photometric measurement that lies at the qth quantile of the library we are using for kernel density estimation in BAYESPHOT, the effective uncertainty our kernel density estimation procedure provides is at a minimum given by the uncertainty in the quantile value x_q calculated via the procedure above. If this uncertainty is larger than the photometric uncertainties in the measurement, then the resolution of the library rather than the accuracy of the measurement will be the dominant source of error.

2.4 SFR_SLUG and CLUSTER_SLUG: Bayesian SFRs and star cluster properties

The SLUG package ships with two Bayesian inference modules based on BAYESPHOT. SFR_SLUG, first described in da Silva et al. (2014) and modified slightly to use the improved computational technique described in the previous section, is a module that infers the posterior probability distribution of the SFR given an observed flux in H α, *GALEX* FUV, or bolometric luminosity, or any combination of the three. The library of models on which it operates (which is included in the SLUG package) consists of approximately 1.8×10^6 galaxies with constant SFRs sampling a range from 10^{-8}–$10^{0.5}$ $\mathrm{M}_\odot$ yr^{-1}, with no extinction; the default bandwidth is 0.1 dex. We refer readers for da Silva et al. (2014) for a further description of the model library. SFR_SLUG can download the default library automatically, and it is also available as a stand-alone download from http://www.slugsps.com/data.

The CLUSTER_SLUG package performs a similar task for inferring the mass, age, and extinction of star clusters. The SLUG models that CLUSTER_SLUG uses consist of libraries of 10^7 SSPs with masses in the range $\log{(M/\mathrm{M}_\odot)} = 2$–8, ages $\log{(T/\mathrm{yr})} = 5 - \log{T_{\max}}$, and extinctions $A_V = 0$–3 mag. The maximum age $T_{\max}$ is either 1 or 15 Gyr, depending on the choice of tracks (see below). The data are sampled uniformly in A_V. In mass the library sampling is $\mathrm{d}N/\mathrm{d}M \propto M^{-1}$ for masses up to 10^4 $\mathrm{M}_\odot$, and as $\mathrm{d}N/\mathrm{d}M \propto M^{-2}$ at higher masses; similarly, the library age sampling is $\mathrm{d}N/\mathrm{d}T \propto T^{-1}$ for $T < 10^8$ yr, and as $\mathrm{d}N/\mathrm{d}T \propto T^{-2}$ at older ages. The motivation for this sampling is that it puts more computational effort at younger ages and low masses, where stochastic variation is greatest. The libraries all use a Chabrier (2005) IMF, and include nebular emission computed using $\phi = 0.73$ and an ionization parameter $\log{\mathcal{U}} = -3$ (see Appendix B). Libraries are available using either Padova or Geneva tracks, with the former having a maximum age of 15 Gyr and the latter a maximum age of 1 Gyr. The Geneva tracks are available using either Milky Way or 'starburst' extinction curves (see Appendix B). The default bandwidth is 0.1 dex in mass and age, 0.1 mag in extinction, and 0.25 mag (corresponding to 0.1 dex in luminosity) in photometry. For each model, photometry is computed for a large range of filters, listed in Table 1. As with SFR_SLUG, CLUSTER_SLUG can automatically download the default library, and the data are also available as a stand-alone download from http://www.slugsps.com/data. Full spectra for the library, allowing the addition of further filters as needed, are also available upon request; they are not provided for web download due to the large file sizes involved.

3 SAMPLE APPLICATIONS

In this section, we present a suite of test problems with the goal of illustrating the different capabilities of SLUG, highlighting the effects of key parameters on SLUG's simulations, and validating the code. Unless otherwise stated, all computations use the Geneva (2013) non-rotating stellar tracks and stellar atmospheres following the STARBURST99 implementation.

3.1 Sampling techniques

As discussed above and in the literature (e.g. Weidner & Kroupa 2006; Haas & Anders 2010; da Silva et al. 2012; Cerviño et al. 2013; Popescu & Hanson 2014), the choice of sampling technique used to generate stellar masses can have significant effects on the distribution of stellar masses and the final light output, even when the underlying distribution being sampled is held fixed. This can have profound astrophysical implications. Variations in sampling technique even for a fixed IMF can produce systematic variations in galaxy colours (e.g. Haas & Anders 2010), nucleosynthetic element yields (e.g. Köppen, Weidner & Kroupa 2007; Haas & Anders 2010), and observational tracers of the SFR (e.g. Weidner, Kroupa & Larsen 2004; Pflamm-Altenburg, Weidner & Kroupa 2007). It is therefore of interest to explore how sampling techniques influence various aspects of stellar populations. We do so as a first demonstration of SLUG's capabilities. Here we consider the problem of populating clusters with stars from an IMF, but our discussion also applies to the case of 'galaxy' simulations. Specifically, we run four sets of 10^4 'cluster' simulations with a target mass of 50 $\mathrm{M}_\odot$ by sampling a Kroupa (2002) IMF with the `STOP_NEAREST`, `STOP_AFTER`, `STOP_BEFORE`, `SORTED_SAMPLING` conditions (see Appendix A1). In the following, we analyse a single timestep at 10^6 yr.

By default, SLUG adopts the `STOP_NEAREST` condition, according to which the final draw from the IMF is added to the cluster only if the inclusion of this last star minimizes the absolute error between the target and the achieved cluster mass. In this case, the cluster mass sometimes exceeds and sometimes falls short of the desired mass. The `STOP_AFTER` condition, instead, always includes the final draw from the IMF. Thus, with this choice, SLUG simulations produce clusters with masses that are always in excess of the target mass. The opposite behaviour is obviously recovered by the `STOP_BEFORE` condition, in which the final draw is always rejected. Finally, for `SORTED_SAMPLING` condition, the final cluster mass depends on the details of the chosen IMF.

Besides this manifest effect of the sampling techniques on the achieved cluster masses, the choice of sampling produces a drastic effect on the distribution of stellar masses, even for a fixed IMF. This is shown in the top panel of Fig. 1, where we display histograms for the mass of the most massive stars within these simulated clusters. One can see that, compared to the default `STOP_NEAREST` condition, the `STOP_AFTER` condition results in more massive stars being included in the simulated clusters. Conversely, the `STOP_BEFORE` and the `SORTED_SAMPLING` undersample the massive end of the IMF. Such a different stellar mass distribution has direct implications for the photometric properties of the simulated stellar populations, especially for wavelengths that are sensitive to the presence of most massive stars. Similar results have previously been obtained by other authors, including Haas & Anders (2010) and Cerviño et al.

Table 1. Filters in the CLUSTER_SLUG library.

Type	Filter name
HST WFC3 UVIS wide	*F225W*, *F275W*, *F336W*, *F360W*, *F438W*, *F475W*, *F555W*, *F606W*, *F775W*, *F814W*
HST WFC3 UVIS medium/narrow	*F547M*, *F657N*, *F658N*
HST WFC3 IR	*F098M*, *F105W*, *F110W*, *F125W*, *F140W*, *F160W*
HST ACS wide	*F435W*, *F475W*, *F555W*, *F606W*, *F625W*, *F775W*, *F814W*
HST ACS medium/narrow	*F550M*, *F658N*, *F660N*
HST ACS HRC	*F330W*
HST ACS SBC	*F125LP*, *F140LP*, *F150LP F165LP*
Spitzer IRAC	3.6, 4.5, 5.8, 8.0
GALEX	FUV, NUV
Johnson–Cousins	*U*, *B*, *V*, *R*, *I*
SDSS	*u*, *g*, *r*, *i*, *z*
2MASS	*J*, *H*, *Ks*

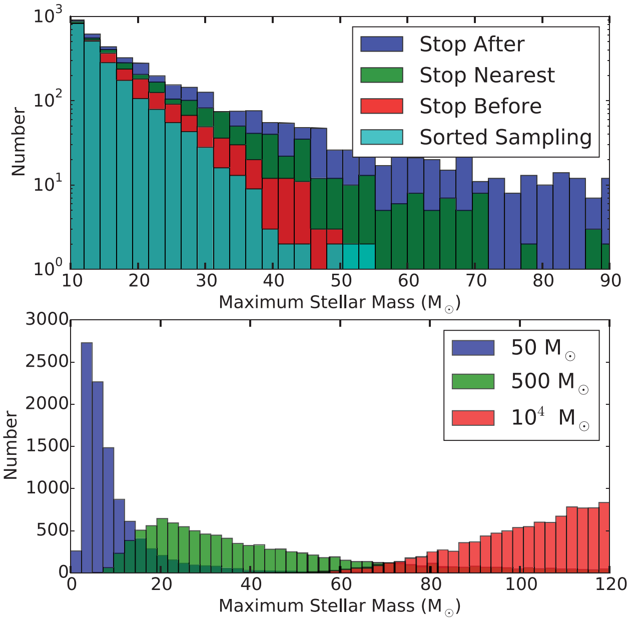

Figure 1. Histograms of the mass of the most massive stars in sets of 10^4 'cluster' simulations, which were performed with different sampling techniques (top panel) and different target cluster masses (bottom panel). In the top panel, four different stop criteria are adopted to sample a Kroupa (2002) IMF for clusters of 50 $M_\odot$. In the bottom panel, the default STOP_NEAREST condition is chosen to sample from a Kroupa (2002) IMF in clusters of three different masses.

(2013), but prior to SLUG no publicly available code has been able to tackle this problem.

The observed behaviour on the stellar mass distribution for a particular choice of sampling stems from a generic mass constraint: clusters cannot be filled with stars that are more massive than the target cluster itself. SLUG does not enforce this condition strictly, allowing for realizations in which stars more massive than the target cluster mass are included. However, different choices of the sampling technique result in a different degree with which SLUG enforces this mass constraint. To further illustrate the relevance of this effect, we run three additional 'cluster' simulations assuming again a Kroupa (2002) IMF and the default `STOP_NEAREST` condition. Each simulation is composed by 10^4 trials, but with a target cluster mass of $M_{\rm cl,\,t} = 50$, 500, and 10^4 $M_\odot$. The bottom panel of Fig. 1 shows again the mass distribution of the most massive star in each cluster. As expected, while massive stars cannot be found in low-mass clusters, the probability of finding at least a star as massive as the IMF upper limit increases with the cluster mass. At the limit of very large masses, a nearly fully sampled IMF is recovered as the mass constraint becomes almost irrelevant. Again, similar results have previously been obtained by Cerviño (2013).

3.2 Stochastic spectra and photometry for varying IMFs

Together with the adopted sampling technique, the choice of the IMF is an important parameter that shapes SLUG simulations. We therefore continue the demonstration of SLUG's capabilities by computing spectra and photometry for SSPs with total masses of 500 $M_\odot$ for three different IMFs. We create 1000 realizations each of such a population at times from 1–10 Myr in intervals of 1 Myr, using the IMFs of Chabrier (2005), Kroupa (2002), and Weidner & Kroupa (2006). The former two IMFs use `STOP_NEAREST` sampling, while the latter uses `SORTED_SAMPLING`, and is otherwise identical to the Kroupa (2002) IMF.

Figs 2–4 show the distributions of spectra and photometry that result from these simulations. Stochastic variations in photometry have been studied previously by a number of authors, going back to Chiosi, Bertelli & Bressan (1988), but to our knowledge no previous authors have investigated similar variations in spectra. The plots also demonstrate SLUG's ability to evaluate the full probability distribution for both spectra and photometric filters, and reveal interesting phenomena that would not be accessible to a non-stochastic SPS code. In particular, Fig. 2 shows that the mean specific luminosity can be orders of magnitude larger than the mode. The divergence is greatest at wavelengths of a few hundred Å at ages ~2–4 Myr, and wavelengths longer than ~5000 Å at 10 Myr. Indeed, at 4 Myr, it is noteworthy that the mean spectrum is actually outside the 10th–90th percentile range. In this particular example, the range of wavelengths at 4 Myr where the mean spectrum is outside the 10th–90th percentile corresponds to energies of ~3 Ryd. For a stellar population 4 Myr old, these photons are produced only by Wolf–Rayet (WR) stars, and most prolifically by extremely hot WC stars. It turns out that a WC star is present ~5 per cent of the time, but these cases are so much more luminous than when a WC star is not present that they are sufficient to drag the mean upwards above the highest luminosities that occur in the ~95 per cent of cases when no WC star is present.

A similar phenomenon is visible in the photometry shown by Fig. 3. In most of the filters the 10th–90th percentile range is an order of magnitude wide, and for the ionizing luminosity and the *HST* UVIS *F814W* at 10 Myr the spread is more than two orders of magnitude, with significant offsets between mean and median

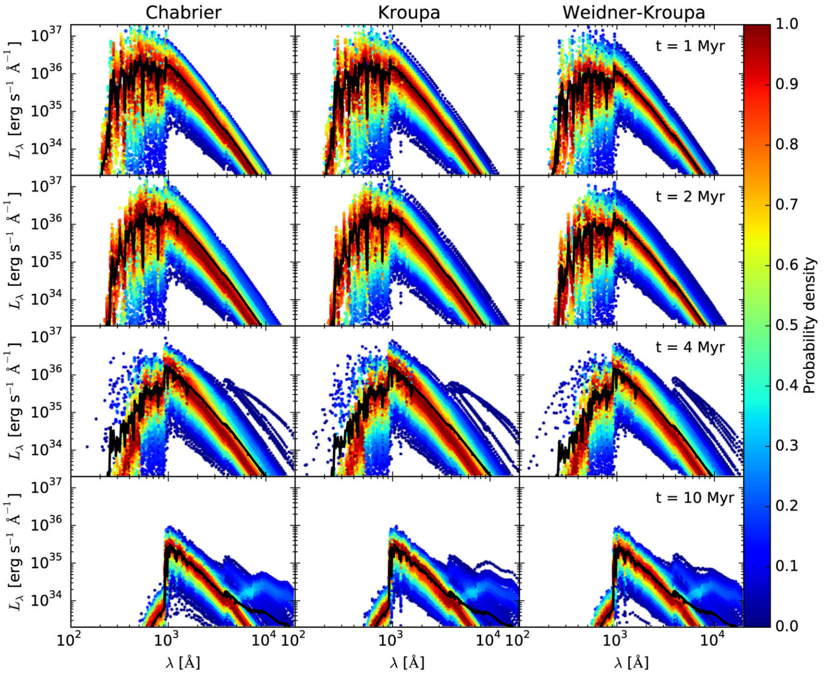

Figure 2. Spectra of SSPs at ages of 1, 2, 4, and 10 Myr (top to bottom rows) for the IMFs of Chabrier (2005, left-hand column), Kroupa (2002, middle column), and Weidner & Kroupa (2006, right-hand column). In each panel, thick black lines indicate the mean, and points show the locations of individual simulations (10 per cent of the simulations, selected at random), with the colour of the point indicating the probability density for the monochromatic luminosity at each wavelength (see colour bar) Probability densities are evaluated independently at each monochromatic luminosity, and are normalized to have a maximum of unity at each wavelength.

indicating a highly asymmetric distribution. Fig. 4, which shows the full distributions for several of the filters, confirms this impression: at early times the cumulative distribution functions are extremely broad, and the ionizing luminosity in particular shows a broad distribution at low $Q(\mathrm{H}^0)$ and then a small number of simulations with large $Q(\mathrm{H}^0)$. Note that the percentile values are generally very well determined by our set of 1000 simulations. Using the method described in Section 2.3.2 to compute the errors on the percentiles, we find that the 68 per cent confidence interval on the 10th, 50th, and 90th percentile values is less than 0.1 dex wide at almost all times, filters, and IMFs. The only exception is at 1 Myr, where the 68 per cent confidence interval on the 10th percentile of ionizing luminosity is ∼0.2–0.3 dex wide.

The figures also illustrate SLUG's ability to capture the 'IGIMF effect' (Weidner & Kroupa 2006) whereby sorted sampling produces a systematic bias towards lower luminosities at short wavelengths and early times. Both the spectra and photometric values for the Weidner & Kroupa (2006) IMF are systematically suppressed relative to the IMFs that use a sampling method that is less biased against high-mass stars (cf. Fig. 1).

3.3 Cluster fraction and CMF

The first two examples have focused on SSPs, namely collections of stars that are formed at coeval times to compose a 'cluster' simulation. Our third example highlights SLUG's ability to simulate composite stellar populations. Due to SLUG's treatment of stellar clustering and stochasticity, simulations of such populations in SLUG differ substantially from those in non-stochastic SPS codes. Unlike in a conventional SPS code, the outcome of a SLUG simulation is sensitive to both the CMF and the fraction f_c of stars formed in clusters. SLUG can therefore be applied to a wide variety of problems in which the fraction of stars formed within clusters or in the field becomes a critical parameter (e.g. Fumagalli et al. 2011).

The relevance of clusters in SLUG simulations arises from two channels. First, because stars form in clusters of finite size, and the interval between cluster formation events is not necessarily large compared to the lifetimes of individual massive stars, clustered star formation produces significantly more variability in the number of massive stars present at any given time than non-clustered star formation. We defer a discussion of this effect to the following section. Here, we instead focus on a second channel by which the CMF and f_c influence the photometric output, which arises due to the method by which SLUG implements mass constraints. As discussed above, a realization of the IMF in a given cluster simulation is the result of the mass constraint imposed by the mass of the cluster, drawn from a CMF, and by the sampling technique chosen to approximate the target mass. Within 'galaxy' simulations, a second mass constraint is imposed on the simulations as the SFH provides an implicit target mass for the galaxy, which in practice constrains each realization of the CMF. As a consequence, all the previous discussion on how the choice of stopping criterion affect the SLUG outputs in cluster simulations applies also to the way with which SLUG approximates the galaxy mass by drawing from the CMF. Through the f_c parameter, the user has control on which level of this hierarchy contributes to the final simulation. In the limit of $f_c = 0$, SLUG removes the intermediate cluster container from the simulations: a galaxy mass is approximated by stars drawn from the IMF, without the constraints imposed by the CMF. Conversely, in the limit of $f_c = 1$, the input SFH constrains the shape of each realization of the CMF, which in turn shapes the mass spectrum of stars within the final outputs. As already noted in Fumagalli et al. (2011), this combined effect resembles in spirit the concept behind the IGIMF theory. However, the SLUG implementation is fundamentally different from the IGIMF, as

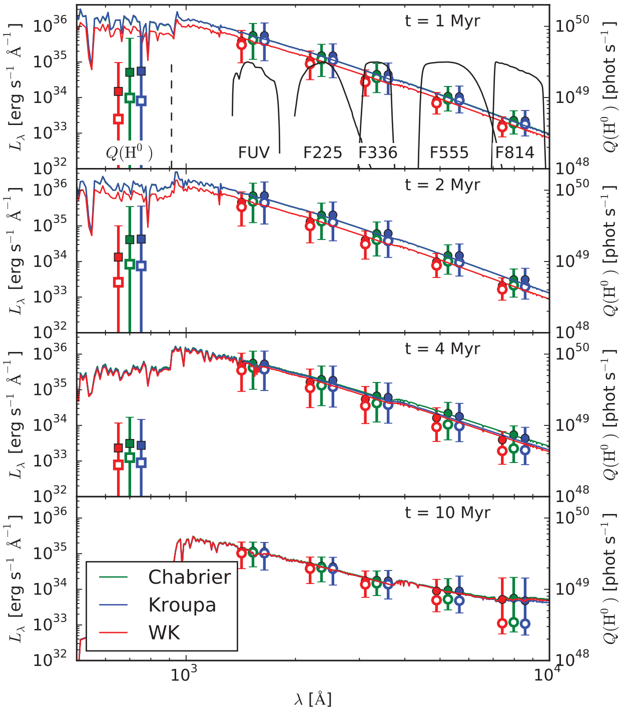

Figure 3. Photometry of SSPs at the same times as shown in Fig. 2. In each panel, the thin lines show the same mean spectra plotted in Fig. 2. Circles with error bars show the specific luminosity L_λ measured in each of the indicated filters: *GALEX* FUV, and *HST* UVIS *F*225*W*, *F*336*W*, *F*555*W*, and *F*814*W*. The abscissae of the green points are placed at the effective wavelength of each filter, with the red and blue offset to either side for clarity. The labelled black curves in the top panel show the filter response functions for each filter. For these points, filled circles indicate the mean value, open circles indicate the median value, and error bars indicate the range from the 10th–90th percentile. The leftmost, square points with error bars show the comparable mean, median, and range for the ionizing luminosity (scale on the right axis).

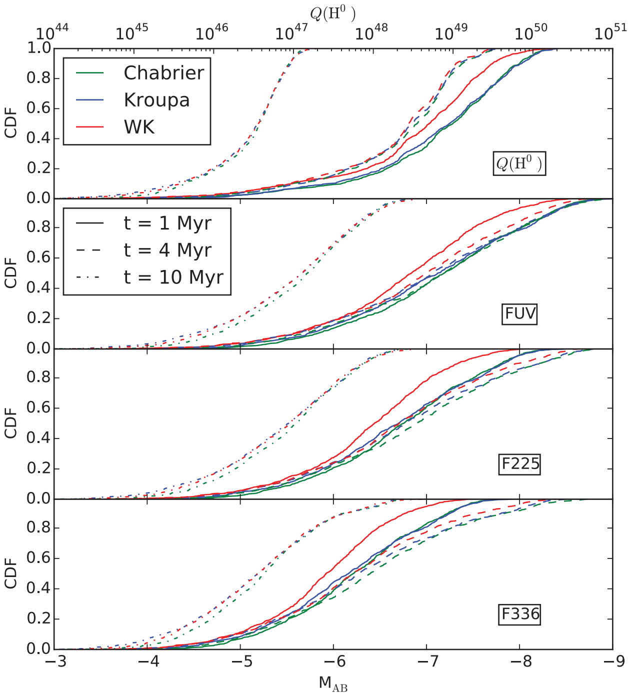

Figure 4. Cumulative distribution functions for ionizing luminosity (top panel) and in the *GALEX* FUV and *HST* UVIS *F*225*W* and *F*336*W* filters (bottom three panels). In each panel, the *x*-axis shows either the ionizing luminosity (for the top panel) or the absolute AB magnitude (bottom three panels) of a 500 M$_\odot$ SSP drawn from a Chabrier (2005, green), Kroupa (2002, blue), or Weidner & Kroupa (2006, red) IMF at ages of 1 Myr (solid), 4 Myr (dashed), and 10 Myr (dot–dashed).

our code does not require any a priori modification of the functional form of the input IMF and CMF, and each realization of these PDFs is only a result of random sampling of invariant PDFs.

To offer an example that better illustrates these concepts, we perform four 'galaxy' simulations with 5000 realizations. This number ensures that ∼10 simulations are present in each mass bin considered in our analysis, thus providing a well-converged distribution. Each simulation follows a single timestep of 2×10^6 yr and assumes a Chabrier (2005) IMF, the default `STOP_NEAREST` condition, and an input SFR of 0.001 M$_\odot$ yr^{-1}. Three of the four simulations further assume a default CMF of the form $\mathrm{d}N/\mathrm{d}M \propto M^{-2}$ between 20–10^7 M$_\odot$, but three different choices of CMF (f_c = 1.0, 0.5, and 0.0). The fourth simulation still assumes $f_\mathrm{c} = 1$ and a $\mathrm{d}N/\mathrm{d}M \propto M^{-2}$ CMF, but within the mass interval 20–100 M$_\odot$ (hereafter the truncated CMF). Results from these simulations are shown in Fig. 5.

By imposing an input SFR of 0.001 M$_\odot$ yr^{-1} for a time 2×10^6 yr, we are in essence requesting that SLUG populates galaxies with 2000 M$_\odot$ worth of stars. However, similarly to the case of cluster simulations, SLUG does not recover the input galaxy mass exactly, but it finds the best approximation based on the chosen stop criteria, the input CMF, and the choice of f_c. This can be seen in the top panel of Fig. 5, where we show the histograms of actual masses from the four simulations under consideration. For $f_\mathrm{c} = 0$, SLUG is approximating the target galaxy mass simply by drawing stars from the IMF. As the typical stellar mass is much less than the desired galaxy mass, the mass constraint is not particularly relevant in these simulations and SLUG reproduces the target galaxy mass quite well, as one can see from the narrow distribution centred around 2000 M$_\odot$. When $f_\mathrm{c} > 0$, however, SLUG tries to approximate the target mass by means of much bigger building blocks, the clusters, thus increasing the spread of the actual mass distributions as seen for instance in the $f_\mathrm{c} = 0.5$ case. For $f_\mathrm{c} > 0$, clusters as massive as 10^7 M$_\odot$ are potentially accessible during draws and, as a consequence of the `STOP_NEAREST` condition, one can notice a wide mass distribution together with a non-negligible tail at very low (including zero) actual galaxy masses. Including a 10^7 M$_\odot$ cluster to approximate a 2000 M$_\odot$ galaxy would in fact constitute a larger error than leaving the galaxy empty! The importance of this effect obviously depends on how massive is the galaxy compared to the upper limit of the CMF. In our example, the choice of an unphysically small 'galaxy' with 2000 M$_\odot$ is indeed meant to exacerbate the relevance of this mass constraint. By inspecting Fig. 5, it is also evident that the resulting galaxy mass distributions are asymmetric, with a longer tail to lower masses. This is due to the fact that SLUG favours small building blocks over massive ones when drawing from a power-law CMF. This means that a draw of a cluster with mass comparable to the galaxy mass will likely occur after a few lower mass clusters have been included in the stellar population. Therefore, massive clusters are more likely to be excluded than retained in a simulation. For this reason, the galaxy target mass represents a limit inferior for the mean actual mass in each distribution.

The middle panel of Fig. 5 shows the total amount of mass formed in clusters after one timestep. As expected, the fraction of mass in

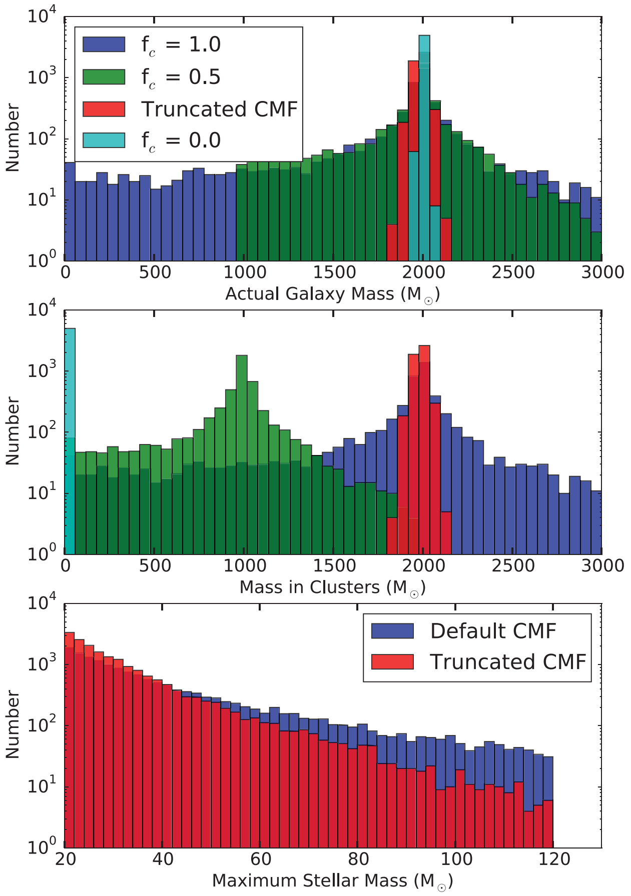

Figure 5. Histograms of the actual galaxy mass (top), total stellar mass in clusters (middle), and mass of the most massive star formed in clusters (bottom) for four different SLUG simulations with 5000 realizations each. The three simulations labelled as f_c consist of the default SLUG CMF and different choices of the fraction of stars formed within clusters. The bottom panel compares instead two simulations with $f_c = 1$, but for two different choices of CMF (SLUG's default and a CMF truncated at 100 M$_\odot$).

clusters versus field stars scales proportionally to f_c, retaining the similar degree of dispersion noted for the total galaxy mass. Finally, by comparing the results of the $f_c = 1$ simulations with the default CMF and the truncated CMF in all the three panels of Fig. 5, one can appreciate the subtle difference that the choice of f_c and CMF have on the output. Even in the case of $f_c = 1$, the truncated CMF still recovers the desired galaxy mass with high precision. Obviously, this is an effect of the extreme choice made for the cluster mass interval, here between 20–100 M$_\odot$. In this case, for the purpose of constrained sampling, the CMF becomes indistinguishable from the IMF, and the simulations of truncated CMF and $f_c = 0$ both recover the target galaxy mass with high accuracy. However, the simulations with truncated CMF still impose a constraint on the IMF, as shown in the bottom panel. In the case of truncated CMF, only clusters up to 100 M$_\odot$ stars are formed, thus reducing the probability of drawing stars as massive as 120 M$_\odot$ from the IMF.

This example highlights how the f_c parameter and the CMF need to be chosen with care based on the problem that one wishes to simulate, as they regulate in a non-trivial way the scatter and the shape of the photometry distributions recovered by SLUG. In passing, we also note that SLUG's ability to handle very general forms for the CMF and IMF makes our code suitable to explore a wide range of models in which the galaxy SFR, CMF, and IMF depend on each other (e.g. as in the IGIMF; Weidner & Kroupa 2006; Weidner, Kroupa & Bonnell 2010).

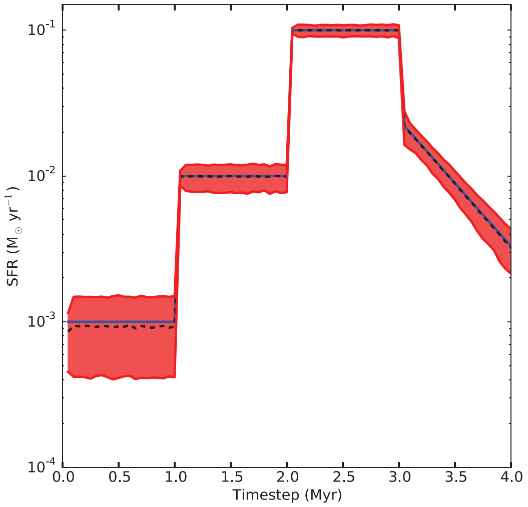

Figure 6. Realizations of the SFH in 5000 SLUG simulations with $f_c = 1$. The input SFH is shown in blue, while the dashed black line and shaded regions show the median, and the first and third quartiles of the distribution.

3.4 Realizations of a given SFH

The study of SFHs in stochastic regimes is receiving much attention in the recent literature, both in the nearby and high-redshift universe (e.g. Boquien, Buat & Perret 2014; Domínguez Sánchez et al. 2014; Kelson 2014). As it can handle arbitrary SFH in input, SLUG is suitable for the analysis of stochastic effects on galaxy SFR.

In previous papers, and particularly in da Silva et al. (2012) and da Silva et al. (2014), we have highlighted the conceptual difference between the input SFH and the outputs that are recovered from SLUG simulations. The reason for such a difference should now be clear from the above discussion: SLUG approximates an input SFH by means of discrete units, either in the form of clusters (for $f_c = 1$), stars (for $f_c = 0$), or a combination of both (for $0 < f_c < 1$). Thus, any smooth input function for the SFH (including a constant SFR) is approximated by SLUG as a series of bursts, that can described conceptually as the individual draws from the IMF or CMF. The effective SFH that SLUG creates in output is therefore an irregular function, which is the result of a superimposition of these multiple bursts. A critical ingredient are the typical time delays with which these bursts are combined, a quantity that is implicitly set by the SFH evaluated in each timestep and by the typical mass of the building blocks used to assemble the simulated galaxies.

A simple example, which also highlights SLUG's flexibility in handling arbitrary SFHs in input, is presented in Fig. 6. For this calculation, we run 5000 SLUG models with default parameters and $f_c = 1$. The input SFH is defined by three segments of constant SFR across three time intervals of 1 Myr each, plus a fourth segment of exponentially decaying SFR with time-scale 0.5 Myr. Fig. 6 shows how the desired SFH is recovered by SLUG on average, but individual models show a substantial scatter about the mean, especially at low SFRs. An extensive discussion of this result is provided in section 3.2 of da Silva et al. (2012).

Briefly, at the limit of many bursts and small time delays (i.e. for high SFRs and/or when SLUG populates galaxies mostly with stars for $f_c \sim 0$), the output SFHs are reasonable approximations of the input SFH. Conversely, for small sets of bursts and for long

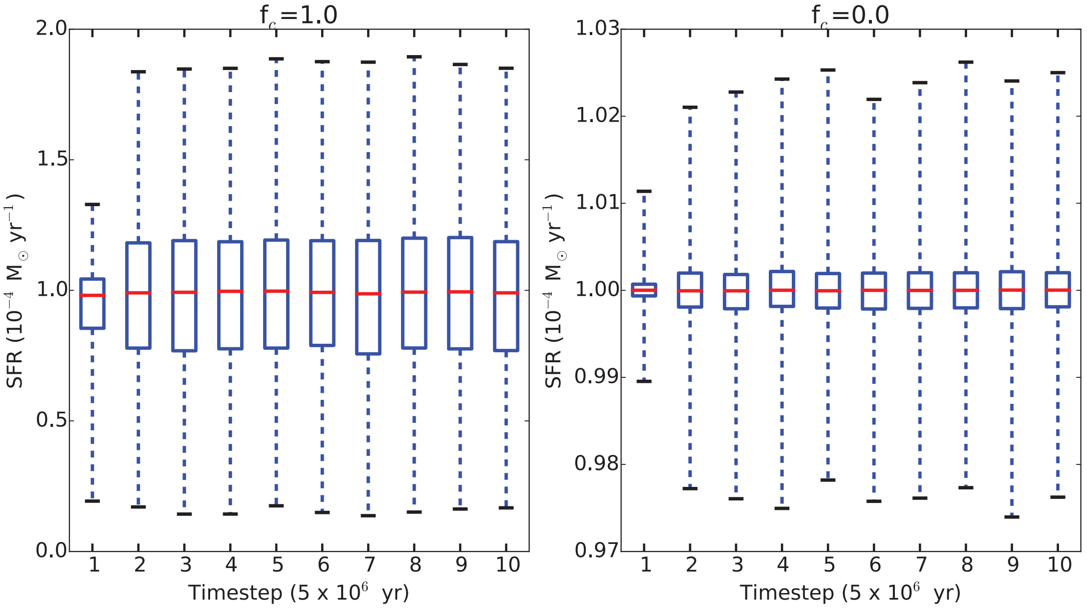

Figure 7. Whisker plots of the recovered SFH from two 'galaxy' simulations of 5000 trials each and input constant SFR of 10^{-4} M$_\odot$ yr^{-1}. The cluster fraction was set to $f_c = 1.0$ (left-hand panel) and $f_c = 0.0$ (right-hand panel). At each timestep, the first and third quartiles are shown by the box plots, with the median marked by the red lines. The 5- and 95-percentiles are also shown by the whiskers.

time delays (i.e. for low SFRs and/or when SLUG populates galaxies mostly with massive clusters for $f_c \sim 1$), the output SFHs are only a coarse representation of the desired input SFH. This behaviour is further illustrated by Fig. 7, in which we show the statistics of the SFHs of 5000 galaxy simulations. These simulations are performed assuming a constant SFR and two choices of fraction of stars formed in clusters, $f_c = 1$ and 0. One can notice that, in both cases, a 'flickering' SFH is recovered, but that a much greater scatter is evident for the $f_c \sim 1$ case when clusters are used to assemble the simulated galaxies.

From this discussion, it clearly emerges that each SLUG simulation will have an intrinsically bursty SFH, regardless to the user-set input, as already pointed out in Fumagalli et al. (2011) and da Silva et al. (2012). It is noteworthy that this fundamental characteristic associated with the discreteness with which star formation occurs in galaxies has also been highlighted by recent analytic and simulation work (e.g. Boquien et al. 2014; Domínguez Sánchez et al. 2014; Kelson 2014; Rodríguez Espinosa et al. 2014). This effect, and the consequences it has on many aspects of galaxy studies including the completeness of surveys or the use of SFR indicators, is receiving great attention particularly in the context of studies at high redshift. SLUG thus provides a valuable tool for further investigation into this problem, particularly because our code naturally recovers a level of burstiness imposed by random sampling, which does not need to be specified a priori as in some of the previous studies.

3.5 Cluster disruption

When performing galaxy simulations, cluster disruption can be enabled in SLUG. In this new release of the code, SLUG handles cluster disruption quite flexibly, as the user can now specify the CLF, which is a PDF from which the lifetime of each cluster is drawn. This implementation generalizes the original cluster disruption method described in da Silva et al. (2012) to handle the wide range of lifetimes observed in nearby galaxies (Adamo et al. 2015). We note however that the SLUG default CLF still follows a power law of index −1.9 between 1 Myr and 1 Gyr as in Fall, Chandar & Whitmore (2009).

To demonstrate the cluster disruption mechanism, we run three simulations of 1000 trials each. These simulations follow the evolution of a burst of 1000 M$_\odot$ between 0 and 1 Myr with a timestep of 5×10^5 yr up to a maximum time of 10 Myr. All stars are formed in clusters. The three simulations differ only for the choice of cluster disruption: one calculation does not implement any disruption law, while the other two assume a CLF in the form of a power law with indices −1.9 and −1 between 1 and 1000 Myr. Results from these calculations are shown in Fig. 8.

The total mass in clusters, as expected, rises till a maximum of 1000 M$_\odot$ at 1 Myr, at which point it remains constant for the non-disruption case, while it declines according to the input power law in the other two simulations. When examining the galaxy bolometric luminosity, one can see that the cluster disruption has no effect on the galaxy photometry. In this example, all stars are formed in clusters and thus all the previous discussion on the mass constraint also applies here. However, after formation, clusters and galaxies are passively evolved in SLUG by computing the photometric properties as a function of time. When a cluster is disrupted, SLUG stops tagging it as a 'cluster' object, but it still follows the contribution that these stars make to the integrated 'galaxy' properties. Clearly, more complex examples in which star formation proceeds both in the field and in clusters following an input SFH while cluster disruption is enabled would exhibit photometric properties that are set by the passive evolution of previously formed stars and by the zero-age main sequence properties of the newly formed stellar populations, each with its own mass constraint.

3.6 Dust extinction and nebular emission

In this section we offer an example of simulations which implement dust extinction and nebular emission in post-processing, two new features offered starting from this release of the code (see Section 2.1.2). Fig. 9 shows the stellar spectra of three SLUG

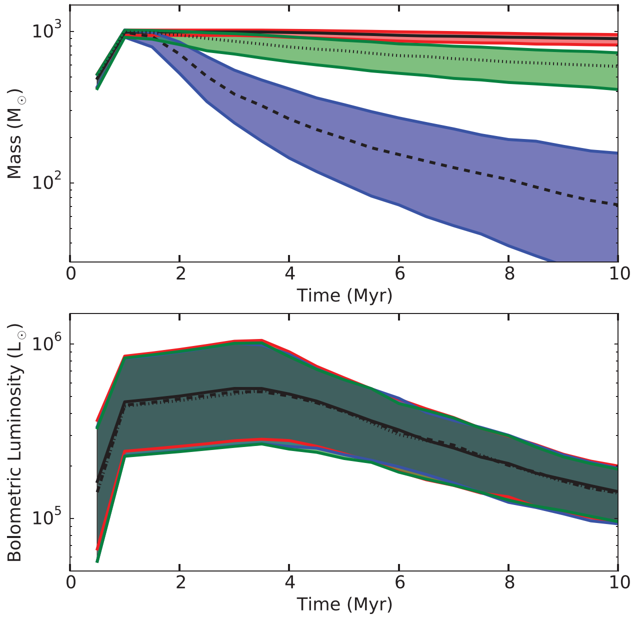

Figure 8. Total mass inside clusters (top) and the galaxy bolometric luminosity (bottom) as a function of time for three SLUG simulations of 1000 trials each. In all cases, 1000 $M_\odot$ of stars are formed in a single burst within 1 Myr from the beginning of the calculation. Simulations without cluster disruption are shown in red, while simulations with cluster disruption enabled according to a power-law CLF of index −1.9 and −1 are shown in blue and green. The black and coloured thick lines show the median, first, and third quartiles of the distributions.

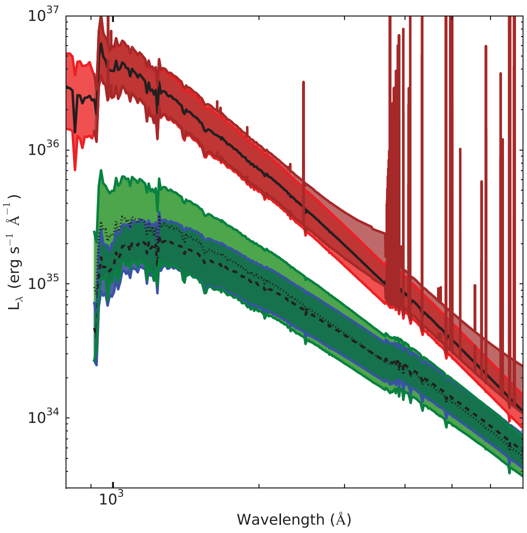

Figure 9. Spectra of four 'galaxy' simulations with SFR of 0.001 $M_\odot$ yr^{-1} evaluated at time of 2×10^6 yr. Each simulation differs for the adopted extinction law or the inclusion of nebular emission. In red (solid line), the intrinsic stellar spectrum is shown, while models with deterministic and probabilistic extinction laws are shown, respectively, in blue (dashed line) and green (dotted line). The solid lines show the first, second, and third quartiles of 5000 realizations. In brown, the range of luminosities (first and third quartiles) with the inclusion of nebular emission is shown.

simulations (in green, blue, and red, respectively) of a galaxy that is forming stars with a SFR of 0.001 $M_\odot$ yr^{-1}. During these simulations, each of 5000 trials, the cluster fraction is set to $f_c = 1$ and photometry is evaluated at 2×10^6 yr. Three choices of extinction law are adopted: the first simulation has no extinction, while the second and third calculations implement the Calzetti et al. (2000) extinction law. In one case, a deterministic uniform screen with $A_V = 1$ mag is applied to the emergent spectrum, while in the other case the value of A_V is drawn for each model from a lognormal distribution with mean 1.0 mag and dispersion of ∼0.3 dex (the SLUG default choice).

As expected, the simulations with a deterministic uniform dust screen closely match the results of the non-dusty case, with a simple wavelength-dependent shift in logarithmic space. For the probabilistic dust model, on the other hand, the simulation results are qualitatively similar to the non-dusty case, but display a much greater scatter due to the varying degree of extinction that is applied in each trial. This probabilistic dust implementation allows one to more closely mimic the case of non-uniform dust extinction, in which different line of sights may be subject to a different degree of obscuration. One can also see how spectra with dust extinction are computed only for $\lambda > 912$ Å. This is not a physical effect, but it is a mere consequence of the wavelength range for which the extinction curves have been specified.

Finally, we also show in Fig. 9 a fourth simulation (brown colour), which is computed including nebular emission with SLUG's default parameters: $\phi = 0.73$ and $\log \mathcal{U} = -3$ (see Appendix B). In this case, the cut-off visible at the ionization edge is physical, as SLUG reprocesses the ionizing radiation into lower frequency nebular emission according to the prescriptions described in Section 2.1.2. One can in fact notice, besides the evident contribution from recombination lines, an increased luminosity at $\lambda \gtrsim 2500$ Å that is a consequence of free–free, bound–free, and two-photon continuum emission.

3.7 Coupling CLOUDY to SLUG simulations

In this section, we demonstrate the capability of the CLOUDY_SLUG package that inputs the results of SLUG simulations into the CLOUDY radiative transfer code. For this, we run 100 realizations of a 'galaxy' simulation following a single timestep of 2×10^6 yr. The input parameters for the simulation are identical to those in Section 3.3. The galaxy is forming stars at a rate of 0.001 $M_\odot$ yr^{-1} with $f_c = 1$. We then pipe the SLUG output into CLOUDY to simulate H II regions in integrated mode, following the method discussed in Section 2.2. In these calculations, we assume the default parameters in CLOUDY_SLUG, and in particular a density in the surroundings of the H II regions of 10^3 cm^{-3}.

Results are shown in Fig. 10. In the top panel, the median of the 100 SLUG spectral energy distributions (SEDs) is compared to the median of the 100 SEDs returned by CLOUDY, which adds the contribution of both the transmitted and the diffuse radiation. As expected, the processed CLOUDY SEDs resemble the input SLUG spectra. Photons at short wavelengths are absorbed inside the H II regions and are converted into UV, optical, and IR photons, which are then re-emitted within emission lines or in the continuum.[6] The nebula-processed spectrum also shows the effects of dust absorption within the H II region and its surrounding shell, which explains why the

[6] We do not show the region below 912 Å in Fig. 10 so that we can zoom-in on the features in the optical region.

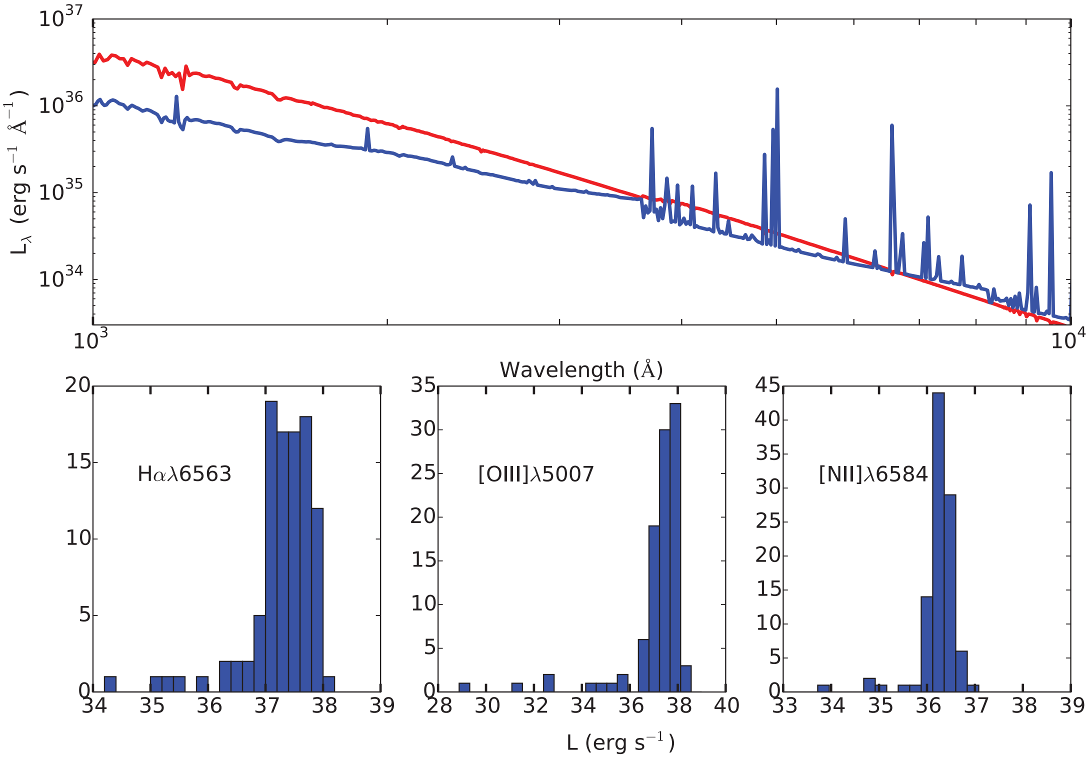

Figure 10. Results of 100 SLUG and CLOUDY simulations for a galaxy with continuous SFR of 0.001 $M_\odot$ yr^{-1} and $f_c = 1$. Top panel: the median SED derived from the 100 SLUG calculations is shown in red, while the median of the CLOUDY SEDs computed in integrated mode is shown in blue. Bottom panels: histograms of the line luminosity for three selected transitions computed by CLOUDY in integrated mode for each input SLUG SED.

nebular spectrum is suppressed below the intrinsic stellar one at short wavelengths.

The bottom panel shows the full distribution of the line fluxes in three selected transitions: H α, [O III] λ5007 and [N II] λ6584. These distributions highlight how the wavelength-dependent scatter in the input SLUG SEDs is inherited by the reprocessed emission lines, which exhibit a varying degree of stochasticity. The long tail of the distribution at low H α luminosity is not well-characterized by the number of simulations we have run, but this could be improved simply by running a larger set of simulations. We are in the process of completing a much more detailed study of stochasticity in line emission (Rendahl, Krumholz & Fumagalli, in preparation).

3.8 Bayesian inference of SFRs

To demonstrate the capabilities of SFR_SLUG, we consider the simple example of using a measured ionizing photon flux to infer the true SFR. We use the library described above and in da Silva et al. (2014), and consider ionizing fluxes which correspond to $\mathrm{SFR}_{Q(\mathrm{H}^0)} = 10^{-5}$, 10^{-3}, and 10^{-1} $M_\odot$ yr^{-1} using an estimate that neglects stochasticity and simply adopts the ionizing luminosity to SFR conversion appropriate for an infinitely sampled IMF and SFH. We then use SFR_SLUG to compute the true posterior probability distribution on the SFR using these measurements; we do so on a grid of 128 points, using photometric errors of 0 and 0.5 dex, and using two different prior probability distributions: one that is flat in log SFR (i.e. $\mathrm{d}p/\mathrm{d}\log\mathrm{SFR} \sim$ constant), and one following the Schechter function distribution of SFRs reported by Bothwell et al. (2011), $\mathrm{d}p/\mathrm{d}\log\mathrm{SFR} \propto \mathrm{SFR}^{\alpha}\exp(-\mathrm{SFR}/\mathrm{SFR}_\star)$, where $\alpha = -0.51$ and $\mathrm{SFR}_\star = 9.2$ $M_\odot$ yr^{-1}.

Fig. 11 shows the posterior PDFs we obtain, which we normalized to have unit integral. Consistent with the results reported by da Silva et al. (2014), at SFRs of $\sim 10^{-1}$ $M_\odot$ yr^{-1}, the main effect of stochasticity is to introduce a few tenths of a dex uncertainty into the SFR determination, while leaving the peak of the probability distribution centred close to the value predicted by the point mass estimate. For SFRs of 10^{-3} or 10^{-5} $M_\odot$ yr^{-1}, the true posterior PDF is very broad, so that even with a 0.5 dex uncertainty on the photometry, the uncertainty on the true SFR is dominated by stochastic effects. Moreover, the peak of the PDF differs from the value given by the point mass estimate by more than a dex, indicating a systematic bias. These conclusions are not new, but we note that the improved computational method described in Section 2.3 results in a significant code speed-up compared to the method presented in da Silva et al. (2014). The time required for SFR_SLUG to compute the full posterior PDF for each combination of $\mathrm{SFR}_{Q(\mathrm{H}^0)}$, photometric error, and prior probability distribution is ~0.2 s on a single core of a laptop (excluding the start-up time to read the library). Thus this method can easily be used to generate posterior probability distributions for large numbers of measurement in times short enough for interactive use.

3.9 Bayesian inference of star cluster properties

To demonstrate the capabilities of CLUSTER_SLUG, we reanalyse the catalogue of star clusters in the central regions of M83 described by Chandar et al. (2010). These authors observed M83 with Wide Field Camera 3 aboard the *HST*, and obtained measurements for ~650 star clusters in the filters *F336W*, *F438W*, *F555W*, *F814W*, and *F657N* (H α). They used these measurements to assign each

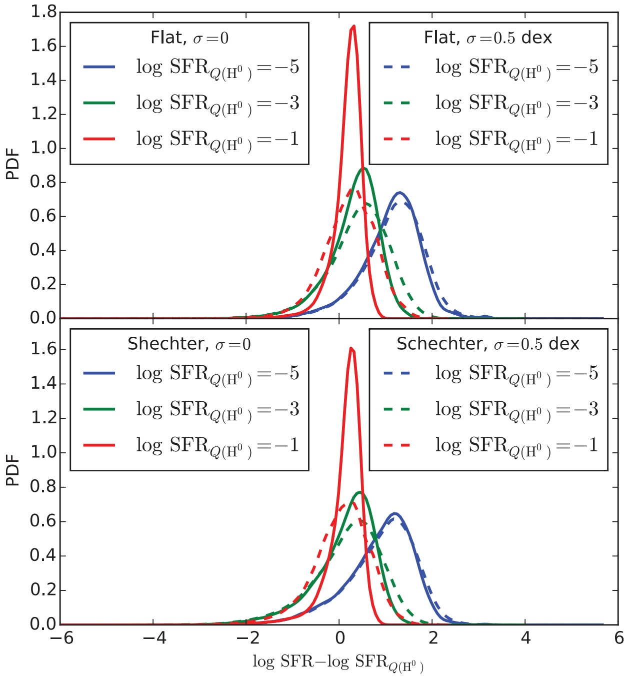

Figure 11. Posterior probability distributions for the logarithmic SFR based on measurements of the ionizing flux, computed with SFR_SLUG. The quantity plotted on the x-axis is the offset between the true log SFR and the value that would be predicted by the 'point mass' estimate where one ignores stochastic effects and simply uses the naive conversion factor between ionizing luminosity and SFR. Thus a value centred around zero indicates that the point mass estimate returns a reasonable prediction for the most likely SFR, while a value offset from zero indicates a systematic error in the point mass estimate. In the top panel, solid curves show the posterior PDF for a flat prior probability distribution and no photometric errors ($\sigma = 0$), with the three colours corresponding to point mass estimates of $\log \mathrm{SFR}_{Q(\mathrm{H}^0)} = -5$, -3, and -1 based on the ionizing flux. The dashed lines show the same central values, but with assumed errors of $\sigma = 0.5$ dex in the measured ionizing flux. In the bottom panel, we show the same quantities, but computed using a Schechter function prior distribution rather than a flat one (see main text for details).

cluster a mass and age by comparing the observed photometry to SSP models using Bruzual & Charlot (2003) models for a twice-solar metallicity population, coupled with a Milky Way extinction law; see Chandar et al. (2010) for a full description of their method. This catalogue has also been reanalysed by Fouesneau et al. (2012) using their stochastic version of PEGAGSE. Their method, which is based on χ^2 minimization over a large library of simulated clusters, is somewhat different than our kernel density-based one, but should share a number of similarities – see Fouesneau & Lançon (2010) for more details on their method. We can therefore compare our results to theirs as well.

We downloaded the Chandar et al. (2010) 'automatic' catalogue from MAST[7] and used CLUSTER_SLUG to compute posterior probability distributions for the mass and age of every cluster for which photometric values were available in all five filters. We used the photometric errors included in the catalogue in this analysis, coupled to the default choice of bandwidth in CLUSTER_SLUG, and a prior probability distribution that is flat in the logarithm of the age and A_V, while varying with mass as $p(\log M) \propto 1/M$. We used our library computed for the Padova tracks and a Milky Way extinction curve, including nebular emission. The total time required for CLUSTER_SLUG to compute the marginal probability distributions of mass and age on grids of 128 logarithmically spaced points each was ~4 s per marginal PDF (~6000 s for two PDFs each on the entire catalogue of 656 clusters), using a single CPU. The computation can be parallelized trivially simply by running multiple instances of CLUSTER_SLUG on different parts of the input catalogue.

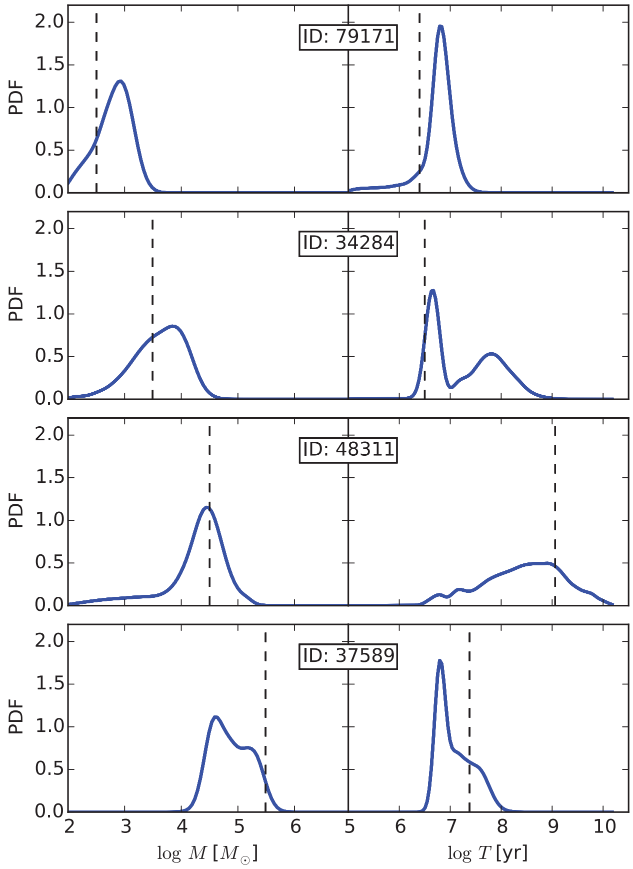

Figure 12. Posterior probability distributions for cluster mass (left) and age (right) for four sample clusters taken from the catalogue of Chandar et al. (2010), as computed by CLUSTER_SLUG. In each panel, the heavy blue line is the CLUSTER_SLUG result, and the thin vertical dashed line is the best fit obtained by Chandar et al. (2010). The ID number printed in each pair of panels is the ID number assigned in the Chandar et al. (2010) catalogue.

In Fig. 12 we show example posterior probabilities distributions for cluster mass and age as returned by CLUSTER_SLUG, and in Fig. 13 we show the median and interquartile ranges for cluster mass and age computed from CLUSTER_SLUG compared to those determined by Chandar et al. (2010).[8] The points in Fig. 13 are colour-coded by the 'photometric distance' between the observed photometric values and the 5th closest matching model in the CLUSTER_SLUG library, where the photometric distance is defined as

$$d = \sqrt{\frac{1}{N_{\mathrm{filter}}} \sum_{i=1}^{N_{\mathrm{filter}}} (M_{i,\mathrm{obs}} - M_{i,\mathrm{lib}})^2}, \tag{29}$$

[7] https://archive.stsci.edu/

[8] The mass and age estimates plotted are from the most up-to-date catalogue maintained by Chandar (private communication).

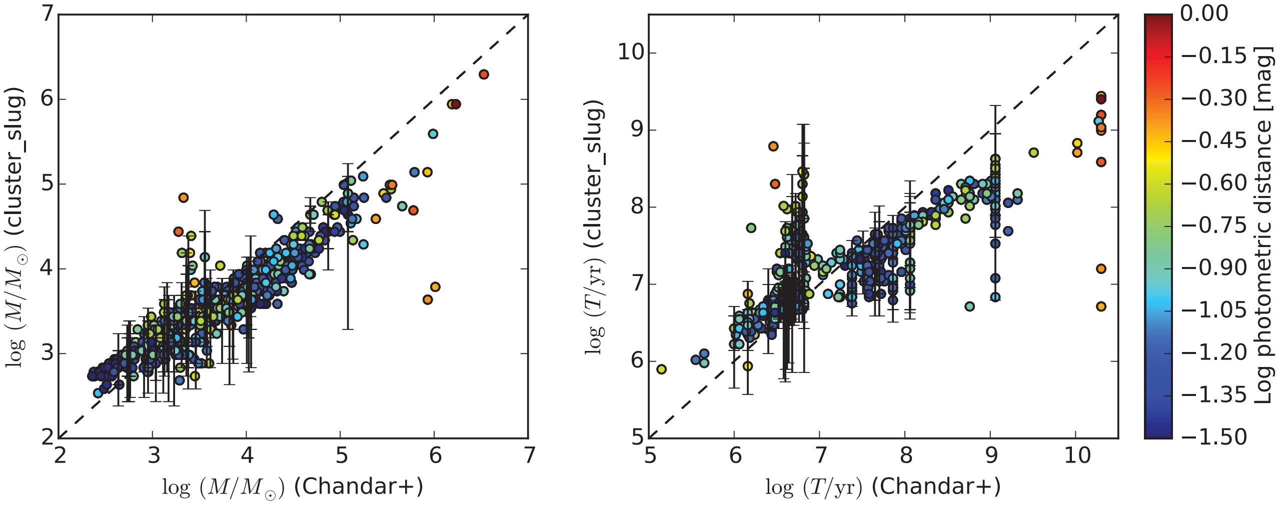

Figure 13. Comparison of cluster masses (left) and ages (right) for clusters in the Chandar et al. (2010) catalogue computed using two different methods. For each cluster, the value shown on the x axis is the best-fitting value reported by Chandar et al. (2010) using their non-stochastic model grids. The values shown on the y-axis are the median masses and ages computed by CLUSTER_SLUG; for every 20th cluster we also show error bars, which indicate the 10th–90th percentile range computed from the CLUSTER_SLUG posterior PDF. The dashed black lines indicate the 1–1 line, i.e. perfect agreement. Points are coloured by the photometric distance (see main text for definition) between the observed photometry for that cluster and the 5th closest match in the CLUSTER_SLUG simulation library.

where $M_{i,\,\mathrm{obs}}$ and $M_{i,\,\mathrm{lib}}$ are the observed magnitude and the magnitude of the CLUSTER_SLUG simulation in filter i, and the sum is over all five filters used in the analysis.

We can draw a few conclusions from these plots. Examining Fig. 12, we see that CLUSTER_SLUG in some cases identifies a single most likely mass or age with a fairly sharp peak, but in other cases identifies multiple distinct possible fits, so that the posterior PDF is bimodal. In these cases the best fit identified by Chandar et al. (2010) usually matches one of the peaks found by CLUSTER_SLUG. The ability to recover bimodal posterior PDFs represents a distinct advantage of CLUSTER_SLUG's method, since it directly indicates cases where there is a degeneracy in possible models.

From Fig. 13, we see that, with the exception of a few catastrophic outliers, the median masses returned by CLUSTER_SLUG are comparable to those found by Chandar et al. (2010). The ages show general agreement, but less so than the masses. For the ages, disagreements come in two forms. First, there is a systematic difference that CLUSTER_SLUG tends to assign younger ages for any cluster for which Chandar et al. (2010) have assigned an age $>10^8$ yr. For many of these clusters the 1–1 line is still within the 10th–90th percentile range, but the 50th percentile is well below the 1–1 line. The second type of disagreement is more catastrophic. We find that there are a fair number of clusters for which Chandar et al. (2010) assign ages of ∼5 Myr, while CLUSTER_SLUG produces ages 1–2 dex larger. Conversely, for the oldest clusters in the Chandar et al. (2010) catalogue, CLUSTER_SLUG tends to assign significantly younger ages.

The differences in ages probably have a number of causes. The fact that our 50th percentile ages tend to be lower than those derived by Chandar et al. (2010) even when the disagreement is not catastrophic is likely a result of the broadness of the posterior PDFs, and the fact that we are exploring the full posterior PDF rather than selecting a single best fit. For example, in the middle two panels of Fig. 12, the peak of the posterior PDF is indeed close to the value assigned by Chandar et al. (2010) However, the PDF is also very broad, so that the 50th percentile can be displaced very significantly from the peak. We note that this also likely explains why our Fig. 13 appears to indicate greater disagreement between stochastic and non-stochastic models than do figs 18 and 20 of Fouesneau et al. (2012), which ostensibly make the same comparison. Fouesneau et al. (2012) identify the peak in the posterior PDF, but do not explore its full shape. When the PDF is broad or double-peaked, this can be misleading.

The catastrophic outliers are different in nature, and can be broken into two categories. The disagreement at the very oldest ages assigned by Chandar et al. (2010) is easy to understand: the very oldest clusters in M83 are globular clusters with substantially subsolar metallicities, while our library is for solar metallicity. Thus our library simply does not contain good matches to the colours of these clusters, as is apparent from their large photometric distances (see more below). The population of clusters for which Chandar et al. (2010) assign ∼5 Myr ages, while the stochastic models assign much older ages, is more interesting. Fouesneau et al. (2012) found a similar effect in their analysis, and their explanation of the phenomenon holds for our models as well. These clusters have relatively red colours and lack strong H α emission, properties that can be matched almost equally well either by a model at an age of ∼5 Myr (old enough for ionization to have disappeared) with relatively low mass and high extinction, or by a model at a substantially older age with lower extinction and higher total mass. This is a true degeneracy, and the stochastic and non-stochastic models tend to break the degeneracy in different directions. The ages assigned to these clusters also tend to be quite dependent on the choice of priors (Krumholz et al., in preparation).

We can also see from Fig. 13 that, for the most part and without any fine tuning, the default CLUSTER_SLUG library does a very good job of matching the observations. As indicated by the colour-coding, for most of the observed clusters, the CLUSTER_SLUG library contains at least five simulations that match the observations to a photometric distance of a few tenths of a magnitude. Quantitatively, 90 per cent of the observed clusters have a match in the library that is within 0.15 mag, and 95 per cent have a match within 0.20 mag; for 5th nearest neighbours, these figures rise to 0.18 and 0.22 mag. There are, however, some exceptions, and these are clusters for which the CLUSTER_SLUG fit differs most dramatically from the Chandar et al. (2010) one. The worst agreement is found for the clusters for which Chandar et al. (2010) assigns the oldest ages, and, as noted above, this is almost certainly a metallicity effect. However, even eliminating these cases there are ∼10 clusters for which the closest match in the cluster_slug library is at a photometric distance of 0.3–0.6 mag. It is possible that these clusters have extinctions $A_V > 3$ and thus outside the range covered by the library, that these are clusters where our failure to make the appropriate aperture corrections makes a large difference, or that the disagreement has some other cause. These few exceptions notwithstanding, this experiment makes it clear that, for the vast majority of real star cluster observations, cluster_slug can return a full posterior PDF that properly accounts for stochasticity and other sources of error such as age–mass degeneracies, and can do so at an extremely modest computational cost.

4 SUMMARY AND PROSPECTS

As telescopes gains in power, observations are able to push to ever smaller spatial and mass scales. Such small scales are often the most interesting, since they allow us to observe fine details of the process by which gas in galaxies is transformed into stars. However, taking advantage of these observational gains will require the development of a new generation of analysis tools that dispense with the simplifying assumptions that were acceptable for processing lower resolution data. The goal of this paper is to provide such a next-generation analysis tool that will allow us to extend SPS methods beyond the regime where stellar IMFs and SFHs are well sampled. The slug code we describe here makes it possible to perform full SPS calculations with nearly arbitrary IMFs and SFHs in the realm where neither distribution is well sampled; the accompanying suite of software tools makes it possible to use libraries of slug simulations to solve the inverse problem of converting observed photometry to physical properties of stellar populations outside the well-sampled regime. In addition to providing a general software framework for this task, bayesphot, we provide software to solve the particular problems of inferring the posterior probability distribution for galaxy SFRs (sfr_slug) and star cluster masses, ages, and extinctions (cluster_slug) from unresolved photometry.

In upcoming work, we will use slug and its capability to couple to cloudy (Ferland et al. 2013) to evaluate the effects of stochasticity on emission line diagnostics such as the BPT diagram (Baldwin, Phillips & Terlevich 1981), and to analyse the properties of star clusters in the new Legacy Extragalactic UV Survey (Calzetti et al. 2015). However, we emphasize that slug, its companion tools, and the pre-computed libraries of simulations we have created are open source software, and are available for community use on these and other problems.

ACKNOWLEDGEMENTS

The authors thank Michelle Myers for assistance, and Aida Wofford, Angela Adamo, and Janice Lee for comments on the manuscript. MRK acknowledges support from *HST* programmes HST-AR-13256 and HST-GO-13364, provided by NASA through a grant from the Space Telescope Science Institute, which is operated by the Association of Universities for Research in Astronomy, Inc., under NASA contract NAS 5-26555. MF acknowledges support by the Science and Technology Facilities Council, grant number ST/L00075X/1. Some of the data presented in this paper were obtained from the Mikulski Archive for Space Telescopes (MAST). STScI is operated by the Association of Universities for Research in Astronomy, Inc., under NASA contract NAS5-26555. Support for MAST for non-*HST* data is provided by the NASA Office of Space Science via grant NNX13AC07G and by other grants and contracts. This work used the UCSC supercomputer Hyades, which is supported by the NSF (award number AST-1229745), and the DiRAC Data Centric system at Durham University, operated by the Institute for Computational Cosmology on behalf of the STFC DiRAC HPC Facility (www.dirac.ac.uk). This equipment was funded by BIS National E-infrastructure capital grant ST/K00042X/1, STFC capital grant ST/H008519/1, and STFC DiRAC Operations grant ST/K003267/1 and Durham University. DiRAC is part of the National E-Infrastructure.

REFERENCES

Adamo A. et al., 2015, submitted
Akima H., 1970, J. Assoc. Comput. Mach., 17, 589
Anders P., Kotulla R., de Grijs R., Wicker J., 2013, ApJ, 778, 138
Andrews J. E. et al., 2013, ApJ, 767, 51
Andrews J. E. et al., 2014, ApJ, 793, 4
Asa'd R. S., Hanson M. M., 2012, MNRAS, 419, 2116
Astropy Collaboration et al., 2013, A&A, 558, A33
Baldwin J. A., Phillips M. M., Terlevich R., 1981, PASP, 93, 5
Boquien M., Buat V., Perret V., 2014, A&A, 571, A72
Bothwell M. S. et al., 2011, MNRAS, 415, 1815
Bouchet P., Lequeux J., Maurice E., Prevot L., Prevot-Burnichon M. L., 1985, A&A, 149, 330
Bruzual G., Charlot S., 2003, MNRAS, 344, 1000
Calzetti D., Armus L., Bohlin R. C., Kinney A. L., Koornneef J., Storchi-Bergmann T., 2000, ApJ, 533, 682
Calzetti D. et al., 2015, AJ, 149, 51
Cerviño M., 2013, New Astron. Rev., 57, 123
Cerviño M., Luridiana V., 2004, A&A, 413, 145
Cervino M., Luridiana V., 2005, preprint (arXiv:astro-ph/0510411)
Cerviño M., Luridiana V., 2006, A&A, 451, 475
Cerviño M., Valls-Gabaud D., 2003, MNRAS, 338, 481
Cerviño M., Román-Zúñiga C., Luridiana V., Bayo A., Sánchez N., Pérez E., 2013, A&A, 553, A31
Chabrier G., 2003, PASP, 115, 763
Chabrier G., 2005, Astrophys. Space Sci. Libr, 327, 41
Chandar R. et al., 2010, ApJ, 719, 966
Charlot S., Bruzual A. G., 1991, ApJ, 367, 126
Chiosi C., Bertelli G., Bressan A., 1988, A&A, 196, 84
Conroy C., Gunn J. E., 2010, ApJ, 712, 833
Conroy C., Gunn J. E., White M., 2009, ApJ, 699, 486
Conroy C., White M., Gunn J. E., 2010, ApJ, 708, 58
Cook D. O. et al., 2012, ApJ, 751, 100
da Silva R. L., Fumagalli M., Krumholz M., 2012, ApJ, 745, 145
da Silva R. L., Fumagalli M., Krumholz M. R., 2014, MNRAS, 444, 3275
Dalcanton J. J. et al., 2012, ApJS, 200, 18
de Meulenaer P., Narbutis D., Mineikis T., Vansevičius V., 2013, A&A, 550, A20
de Meulenaer P., Narbutis D., Mineikis T., Vansevičius V., 2014, A&A, 569, A4
de Meulenaer P., Narbutis D., Mineikis T., Vansevičius V., 2015, A&A, 574, A66
Domínguez Sánchez H. et al., 2014, MNRAS, 441, 2
Dopita M. A., Groves B. A., Sutherland R. S., Binette L., Cecil G., 2002, ApJ, 572, 753
Draine B. T., 2011, Physics of the Interstellar and Intergalactic Medium. Princeton Univ. Press, Princeton, NJ
Ekström S. et al., 2012, A&A, 537, A146
Ercolano B., Storey P. J., 2006, MNRAS, 372, 1875
Fall S. M., Chandar R., Whitmore B. C., 2009, ApJ, 704, 453
Ferland G. J. et al., 2013, Rev. Mex. Astron. Astrofis., 49, 137
Fioc M., Rocca-Volmerange B., 1997, A&A, 326, 950
Fitzpatrick E. L., 1999, PASP, 111, 63

Foreman-Mackey D., Hogg D. W., Lang D., Goodman J., 2013, PASP, 125, 306
Forero-Romero J. E., Dijkstra M., 2013, MNRAS, 428, 2163
Fouesneau M., Lançon A., 2010, A&A, 521, A22
Fouesneau M., Lançon A., Chandar R., Whitmore B. C., 2012, ApJ, 750, 60
Fouesneau M. et al., 2014, ApJ, 786, 117
Fumagalli M., da Silva R. L., Krumholz M. R., 2011, ApJ, 741, L26
Georgy C. et al., 2013, A&A, 558, A103
Girardi L., Bressan A., Bertelli G., Chiosi C., 2000, A&AS, 141, 371
Haas M. R., Anders P., 2010, A&A, 512, A79
Hillier D. J., Miller D. L., 1998, ApJ, 496, 407
Jones T., Ellis R. S., Richard J., Jullo E., 2013a, ApJ, 765, 48
Jones T. A., Ellis R. S., Schenker M. A., Stark D. P., 2013b, ApJ, 779, 52
Kelson D. D., 2014, ApJ, preprint (arXiv:1406.5191)
Köppen J., Weidner C., Kroupa P., 2007, MNRAS, 375, 673
Kroupa P., 2002, Science, 295, 82
Krumholz M. R., 2014, Phys. Rep., 539, 49
Krumholz M. R., Matzner C. D., 2009, ApJ, 703, 1352
Krumholz M. R., McKee C. F., Tumlinson J., 2009, ApJ, 693, 216
Lada C. J., Lada E. A., 2003, ARA&A, 41, 57
Landini M., Natta A., Salinari P., Oliva E., Moorwood A. F. M., 1984, A&A, 134, 284
Leitherer C. et al., 1999, ApJS, 123, 3
Leitherer C., Ortiz Otálvaro P. A., Bresolin F., Kudritzki R.-P., Lo Faro B., Pauldrach A. W. A., Pettini M., Rix S. A., 2010, ApJS, 189, 309
Leitherer C., Ekström S., Meynet G., Schaerer D., Agienko K. B., Levesque E. M., 2014, ApJS, 212, 14
Lejeune T., Cuisinier F., Buser R., 1997, A&AS, 125, 229
Leroy A. K. et al., 2013, AJ, 146, 19
McKee C. F., Williams J. P., 1997, ApJ, 476, 144
Maíz Apellániz J., 2009, ApJ, 699, 1938
Meynet G., Maeder A., Schaller G., Schaerer D., Charbonnel C., 1994, A&AS, 103, 97
Murray N., Rahman M., 2010, ApJ, 709, 424
Nussbaumer H., Schmutz W., 1984, A&A, 138, 495
Osterbrock D. E., 1989, Astrophysics of Gaseous Nebulae and Active Galactic Nuclei. University Science Books, Mill Valley, CA
Pauldrach A. W. A., Hoffmann T. L., Lennon M., 2001, A&A, 375, 161
Pflamm-Altenburg J., Weidner C., Kroupa P., 2007, ApJ, 671, 1550
Popescu B., Hanson M. M., 2009, AJ, 138, 1724
Popescu B., Hanson M. M., 2010a, ApJ, 713, L21
Popescu B., Hanson M. M., 2010b, ApJ, 724, 296
Popescu B., Hanson M. M., 2014, ApJ, 780, 27
Popescu B., Hanson M. M., Elmegreen B. G., 2012, ApJ, 751, 122
Rodríguez Espinosa J. M. et al., 2014, MNRAS, 444, L68
Salpeter E. E., 1955, ApJ, 121, 161
Schaller G., Schaerer D., Meynet G., Maeder A., 1992, A&AS, 96, 269
Smith L. J., Norris R. P. F., Crowther P. A., 2002, MNRAS, 337, 1309
Storey P. J., Hummer D. G., 1995, MNRAS, 272, 41
Vassiliadis E., Wood P. R., 1993, ApJ, 413, 641
Vázquez G. A., Leitherer C., 2005, ApJ, 621, 695
Verdolini S., Yeh S. C. C., Krumholz M. R., Matzner C. D., Tielens A. G. G. M., 2013, ApJ, 769, 12
Weidner C., Kroupa P., 2006, MNRAS, 365, 1333
Weidner C., Kroupa P., Larsen S. S., 2004, MNRAS, 350, 1503
Weidner C., Kroupa P., Bonnell I. A. D., 2010, MNRAS, 401, 275
Weidner C., Kroupa P., Pflamm-Altenburg J., 2014, MNRAS, 441, 3348
Yeh S. C. C., Matzner C. D., 2012, ApJ, 757, 108
Yeh S. C. C., Verdolini S., Krumholz M. R., Matzner C. D., Tielens A. G. G. M., 2013, ApJ, 769, 11

APPENDIX A: IMPLEMENTATION DETAILS AND CAPABILITIES

Here we describe some details of SLUG's current implementation. These capabilities may be expanded in future versions, but we include this description here both to demonstrate the code's flexibility, and to discuss some subtleties that may be of use to those interested in implementing similar codes.

A1 Probability distribution functions

SLUG uses a large number of PDFs. In particular, the IMF, the CMF, and the SFH are all required to be PDFs, and the extinction, output time, and (for simulations of SSPs) can optionally be described by PDFs as well. In SLUG, PDFs can be specified as a sum of an arbitrary number of piecewise continuous segments,

$$\frac{\mathrm{d}p}{\mathrm{d}x} = n_1 f_1(x; x_{1,a}, x_{1,b}) + n_2 f_2(x; x_{2,a}, x_{2,b}) + \cdots, \tag{A1}$$

where the normalizations n_i for each segment are free parameters, as are the parameters $x_{i,\,a}$ and $x_{i,\,b}$ that denote the lower and upper limits for each segment (i.e. the function $f_i(x)$ is non-zero only in the range $x \in [x_{i,\,a}, x_{i,\,b}]$). The functions $f_i(x)$ can take any of the functional forms listed in Table A1, and the software is modular so that additional functional forms can be added very easily. In the most common cases, the segment limits and normalizations will be chosen so that the segments are contiguous with one another and the overall PDF continuous, i.e. $x_{i,\,a} = x_{i,\,b}$ and $n_i f_i(x_{i,\,b}) = n_{i\,+\,1} f_{i\,+\,1}(x_{i\,+\,1,\,a})$. However, this is not a required condition, and the limits, normalizations, and number of segments can be varied arbitrarily. In particular, segments are allowed to overlap or to be discontinuous (as they must in the case of δ function segments); thus for example one could treat the SFH of a galaxy as a sum of two components, one describing the bulge and one describing the disc. SLUG ships with the following IMFs pre-defined for user convenience: Salpeter (1955), Kroupa (2002), Chabrier (2003), and Chabrier (2005).

When drawing a finite total mass, in addition to the mass distribution, one must also specify a sampling method to handle the fact that one will not be able to hit the target mass perfectly when drawing discrete objects. SLUG allows users to choose a wide range of methods for this purpose, which we describe briefly here following the naming convention used in the code. The majority of these methods are described in Haas & Anders (2010).

(i) `STOP_NEAREST`: draw from the IMF until the total mass of the population exceeds M_{target}. Either keep or exclude the final star drawn depending on which choice brings the total mass closer to the target value. Unless a different scheme is deemed necessary, this is the preferred and default choice of SLUG, as this sampling technique ensures that the stochastic solution converges towards the deterministic one at the limit of sufficiently large M_{target}.

(ii) `STOP_BEFORE`: same as `STOP_NEAREST`, but the final star drawn is always excluded.

(iii) `STOP_AFTER`: same as `STOP_NEAREST`, but the final star drawn is always kept.

Table A1. Functional forms for PDF segments in SLUG.

Name	Functional form, $f(x)$	Parameters[a]
`delta`	$\delta(x - x_0)$	x_0[b]
`exponential`	e^{-x/x_*}	x_*
`lognormal`	$x^{-1}\exp\left[-(\ln x/x_0)^2/(2s^2)\right]$	x_0, s
`normal`	$\exp\left[-(x - x_0)^2/(2s^2)\right]$	x_0, s
`powerlaw`	x^p	p
`schechter`	$x^p\mathrm{e}^{-x/x_*}$	p, x_*

[a] In addition to the segment-specific parameters listed, all segments also allow the upper and lower cut-offs x_a and x_b as free parameters. [b] For `delta` segments, we require that $x_a = x_b = x_0$.

(iv) STOP_50: same as STOP_NEAREST, but keep or exclude the final star with 50 per cent probability regardless of which choice gets closer to the target.

(v) NUMBER: draw exactly $N = M_{\rm target}/\langle M\rangle$ objects, where $\langle M\rangle$ is the expectation value for the mass of an object produced by a single draw, and the value of N is rounded to the nearest integer. Note that this method can be used to handle the case of characterizing a population as containing a particular number of objects as opposed to a particular total mass, simply by choosing $M_{\rm target} = N\langle M\rangle$.

(vi) POISSON: draw exactly N objects, where the value of N is chosen from a Poisson distribution with expectation value $\langle N\rangle = M_{\rm target}/\langle M\rangle$

(vii) SORTED_SAMPLING:[9] this method was introduced by Weidner & Kroupa (2006). In it, one first draws exactly $N = M_{\rm target}/\langle M\rangle$ objects as in the NUMBER method. If the resulting total mass $M_{\rm pop}$ is less than $M_{\rm target}$, the procedure is repeated recursively using a target mass $M_{\rm target} - M_{\rm pop}$ until $M_{\rm pop} > M_{\rm target}$. Finally, one sorts the resulting list of objects from least to most massive, and then keeps or removes the final, most massive using a STOP_NEAREST policy.

Finally, we note two limitations in our treatment of the IMF. First, while SLUG allows a great deal of flexibility in its treatment of PDFs, it requires that the various PDFs that appear in the problem (IMF, CMF, etc.) be separable, in the sense that the one cannot depend on the other. Thus for example it is not presently possible to have an IMF that varies systematically over the SFH of a simulation. Secondly, while the IMF can extend to whatever mass is desired, the ability of the code to calculate the light output depends on the availability of stellar evolution tracks extending up to the requisite mass. The set of tracks available in the current version of SLUG (Appendix A2) does not extend above 120 $\mathrm{M}_{\odot}$.

A2 Tracks and atmospheres

The evolutionary tracks used by SLUG consist of a rectangular grid of models for stars' present day-mass, luminosity, effective temperature, and surface abundances at a series of times for a range of initial masses; the times at which the data are stored are chosen to represent equivalent evolutionary stages for stars of different starting masses, and thus the times are not identical from one track to the next. SLUG uses the same options for evolutionary tracks as STARBURST99 (Leitherer et al. 1999, 2010, 2014; Vázquez & Leitherer 2005), and in general its treatment of tracks and atmospheres clones that of STARBURST99 except for minor differences in the numerical schemes used for interpolation and numerical integration (see below). In particular, SLUG implements the latest Geneva models for non-rotating and rotating stars (Ekström et al. 2012; Georgy et al. 2013), as well as earlier models from the Geneva and Padova groups (Schaller et al. 1992; Meynet et al. 1994; Girardi et al. 2000); the latter can also include a treatment of thermally pulsing AGB stars from (Vassiliadis & Wood 1993). The Geneva models are optimized for young stellar populations and likely provide the best-available implementation for them, but they have a minimum mass of 0.8 $\mathrm{M}_{\odot}$, and do not include thermally pulsing - asymptotic giant branch stars, so they become increasingly inaccurate at ages above $\sim 10^8$ yr. The Padova tracks should be valid up to the $\sim$15 Gyr age of the Universe, but are less accurate than the Geneva ones at the earliest ages. See Vázquez & Leitherer (2005) for more discussion. Models are available at a range of metallicities; at present the latest Geneva tracks are available at $Z = 0.014$ and 0.002, the older Geneva tracks are available at $Z = 0.001$, 0.004, 0.008, 0.020, and 0.040, while the Padova tracks are available at $Z = 0.0004$, 0.004, 0.008, 0.020, and 0.050.

SLUG interpolates on the evolutionary tracks using a somewhat higher order version of the isochrone synthesis technique (Charlot & Bruzual 1991) adopted in most other SPS codes. The interpolation procedure is as follows: first, SLUG performs Akima (1970) interpolation in both the mass and time directions for all variables stored on the tracks; interpolations are done in log–log space. Akima interpolation is a cubic spline method with an added limiter that prevents ringing in the presence of outliers; it requires five points in 1D. For tracks where fewer than five points are available, SLUG reverts to linear interpolation. To generate an isochrone, the code interpolates every mass and time track to the desired time, and then uses the resulting points to generate a new Akima interpolation at the desired time.

Note that the choice of interpolation method does not affect most quantities predicted by SPS methods, but it does affect those that are particularly sensitive to discontinuities in the stellar evolution sequence. For example, the production rate of He^{+}-ionizing photons is particularly sensitive to the presence of WR stars, and thus to the interpolation method used to determine the boundary in mass and time of the WR region. We have conducted tests comparing SLUG run with stochasticity turned off to STARBURST99, which uses quadratic interpolation, and find that the spectra produced are identical to a few percent at all wavelengths, with the exception of wavelengths corresponding to photon energies above a few Rydberg at ages of $\sim$4 Myr. Those differences trace back to differences in determining which stars are WR stars, with SLUG's Akima spline giving a slightly different result that STARBURST99's quadratic one. For a more extensive discussion of interpolation uncertainties in SPS models, see Cervino & Luridiana (2005).

Stellar atmospheres are also treated in the same manner as is in STARBURST99 (Smith, Norris & Crowther 2002). By default, stars classified as WR stars based on their masses, effective temperatures, and surface abundances are handled using CMFGEN models (Hillier & Miller 1998), those classified as O and B stars are handled using WM-Basic models (Pauldrach, Hoffmann & Lennon 2001), and the balance are treated using Kurucz atmospheres as catalogued by Lejeune, Cuisinier & Buser (1997, referred to as the BaSeL libraries). Different combinations of the BaSeL, Hillier & Miller (1998), and Pauldrach et al. (2001) atmospheres are also possible.

APPENDIX B: MODELING NEBULAR EMISSION AND EXTINCTION

Here we describe SLUG's model for computing nebular emission and dust extinction.

B1 Nebular continuum and hydrogen lines

As described in the main text, the nebular emission rate can be written as $L_{\lambda,\,\rm neb} = \gamma_{\rm neb}\phi Q(\mathrm{H}^0)/\alpha^{(\mathrm{B})}(T)$. SLUG computes $\alpha^{(\mathrm{B})}(T)$ using

[9] This method replaces the IGIMF method implemented in the earlier version of SLUG (da Silva et al. 2012), which was based on the Weidner et al. (2010) version of the integrated galactic IMF (IGIMF) model. In the Weidner et al. (2010) formulation, the upper cut-off of the IMF in a star cluster depends explicitly (rather than simply due to size-of-sample effects) on the total mass of the cluster. This model has been fairly comprehensively excluded by observations since the original SLUG code was developed (Fumagalli et al. 2011; Andrews et al. 2013, 2014), and Weidner, Kroupa & Pflamm-Altenburg (2014) advocated dropping that formulation of the IGIMF in favour of the earlier Weidner & Kroupa (2006) one. See Krumholz (2014) for a recent review discussing the issue.

the analytic fit provided by Draine (2011, his equation 14.6), and adopts a fiducial value of ϕ following McKee & Williams (1997). However, the user is free to alter ϕ. Similarly, the temperature T can either be user-specified as a fixed value, or can be set automatically from the tabulated CLOUDY data (see Appendix B2).

SLUG computes the nebular emission coefficient as

$$\gamma_{\lambda,\rm neb} = \gamma_{\rm ff}^{\rm (H)} + \gamma_{\rm bf}^{\rm (H)} + \gamma_{\rm 2p}^{\rm (H)} + \sum_{n<n'} \alpha_{nn'}^{\rm eff,(B),(H)} E_{nn'}^{\rm (H)} \\ + x_{\rm He}\gamma_{\rm ff}^{\rm (He)} + x_{\rm He}\gamma_{\rm bf}^{\rm (He)} + \sum_{i} \gamma_{i,\rm line}^{\rm (M)}. \qquad \rm (B1)$$

The terms appearing in this equation are the helium abundance relative to hydrogen $x_{\rm He}$, the H$^+$ and He$^+$ free–free emission coefficients $\gamma_{\rm ff}^{\rm (H)}$ and $\gamma_{\rm ff}^{\rm (He)}$, the H and He bound–free emission coefficients $\gamma_{\rm bf}^{\rm (H)}$ and $\gamma_{\rm bf}^{\rm (He)}$, the H two-photon emission coefficient $\gamma_{\rm 2p}^{\rm (H)}$, the effective emission rates for H recombination lines $\alpha_{nn'}^{\rm eff,(B),(H)}$ corresponding to transitions between principal quantum numbers n and n', the energy differences $E_{nn'}$ between these two states, and the emission coefficient for various metal lines (including He lines) $\gamma_{i,\rm line}^{\rm (M)}$.

We compute each of these quantities as follows. For the H and He free–free emission coefficients, $\gamma_{\rm ff}^{\rm (H)}$ and $\gamma_{\rm ff}^{\rm (He)}$, we use the analytic approximation to the free–free Gaunt factor given by Draine (2011, his equation 10.8). We obtain the corresponding bound–free emission coefficients, $\gamma_{\rm bf}^{\rm (H)}$ and $\gamma_{\rm bf}^{\rm (He)}$, by interpolating on the tables provided by Ercolano & Storey (2006). We obtain the effective case B recombination rate coefficients, $\alpha_{nn'}^{\rm eff,(B),(H)}$, by interpolating on the tables provided by Storey & Hummer (1995). In practice, the sum includes all transitions for which the upper principal quantum number $n' \leq 25$. We compute hydrogen two-photon emission via

$$\gamma_{\rm 2p}^{\rm (H)} = \frac{hc}{\lambda^3} I({\rm H}^0) \alpha_{\rm 2s}^{\rm eff,(B),(H)} \frac{1}{1 + n_{\rm H}/n_{\rm 2s,crit}} P_\nu, \qquad \rm (B2)$$

where $\alpha_{\rm 2s}^{\rm eff,(B),(H)}$ is the effective recombination rate to the 2s state of hydrogen in case B, interpolated from the tables of Storey & Hummer (1995), $n_{\rm 2s,\,crit}$ is the critical density for the 2s–1s transition, and P_ν is the hydrogen two-photon frequency distribution, computed using the analytic approximation of Nussbaumer & Schmutz (1984). The critical density in turn is computed as

$$n_{\rm 2s,crit} = \frac{A_{\rm 2s-1s}}{q_{\rm 2s-2p,p} + (1 + x_{\rm He}) q_{\rm 2s-2p,e}}, \qquad \rm (B3)$$

where $A_{\rm 2s-1s} = 8.23\ {\rm s}^{-1}$ is the effective Einstein coefficient A for the 2s–1s transition (Draine 2011, section 14.2.4), and $q_{\rm 2s-2p,\,p}$ and $q_{\rm 2s-2p,\,e}$ are the rate coefficients for hydrogen 2s–2p transitions induced by collisions with free protons and free electrons, respectively. We take these values from Osterbrock (1989, table 4.10).

B2 Non-hydrogen lines

Metal line emission is somewhat trickier to include. The emission coefficients for metal lines can vary over multiple orders of magnitude depending on H II region properties such as the metallicity, density, and ionization parameter, as well as the shape of the ionizing spectrum. Fully capturing this variation in a tabulated form suitable for fast computation is not feasible, so we make a number of assumptions to reduce the dimensionality of the problem. We consider only uniform-density H II regions with an inner wind bubble of negligible size – in the terminology of Yeh & Matzner (2012), these are classical Strömgren spheres as opposed to wind- or radiation-confined shells. To limit the number of possible spectra we consider, we also consider only spectral shapes corresponding to populations that fully sample the IMF. Thus while our fast estimate still captures changes in the strength of line emission induced by stochastic fluctuations in the overall ionizing luminosity, it does not capture the additional fluctuations that should result from the shape of the ionizing spectrum. A paper studying the importance of these secondary fluctuations is in preparation.

With these choices, the properties of the H II region are to good approximation fully characterized by three parameters: stellar population age, metallicity, and ionization parameter. To sample this parameter space, we use CLOUDY (Ferland et al. 2013) to compute the properties of H II regions on a grid specified as follows.

(i) We consider input spectra produced by using SLUG to calculate the spectrum of a 10^3 M$_\odot$ mono-age stellar population with a Chabrier (2005) IMF and each of our available sets of tracks (see Appendix A2), at ages from 0–10 Myr at intervals of 0.2 Myr. We also consider the spectrum produced by continuous star formation at $\dot{M}_* = 10^{-3}$ M$_\odot$ yr^{-1} over a 10 Myr interval. Note that the mass and SFR affect only the normalization of the spectrum, not its shape.

(ii) For each set of tracks, we set the H II region metallicity relative to solar equal to that of the tracks. Formally, we adopt CLOUDY's 'H II region' abundances case to set the abundance pattern, and then scale the abundances of all gas and grain components by the metallicity of the tracks relative to solar.

(iii) The ionization parameter, which gives the photon to baryon ratio in the H II region, implicitly specifies the density. To complete this specification, we must choose where to measure the ionization parameter, since it will be high near the stars, and will fall off towards the H II region outer edge. To this end, we define our ionization parameter to be the volume-averaged value, which we compute in the approximation whereby the ionizing photon luminosity passing through the shell at radius r is (Draine 2011, section 15.3)

$$Q({\rm H}^0, r) = Q({\rm H}^0) \left[1 - \left(\frac{r}{r_{\rm s}}\right)^3\right]. \qquad \rm (B4)$$

Here $Q({\rm H}^0)$ is the ionizing luminosity of stars at $r = 0$ and

$$r_{\rm s} = \left(\frac{3Q({\rm H}^0)}{4\pi\alpha^{\rm (B)} n_{\rm H}^2}\right)^{1/3} \qquad \rm (B5)$$

is the classical Strömgren radius, which we compute using $\alpha^{\rm (B)}$ evaluated at a temperature of 10^4 K. With these approximations, the volume-averaged ionization parameter is

$$\langle \mathcal{U} \rangle = \frac{3}{4\pi r_{\rm s}^3} \int_0^{r_{\rm s}} 4\pi r^2 \left(\frac{Q({\rm H}^0)}{4\pi r^2 c n_{\rm H}}\right) \left[1 - \left(\frac{r}{r_{\rm s}}\right)^3\right] {\rm d}r \\ = \left[\frac{81 \left(\alpha^{\rm (B)}\right)^2 n_{\rm H} Q({\rm H}^0)}{288\pi c^3}\right]^{1/3}. \qquad \rm (B6)$$

Thus a choice of $\langle \mathcal{U} \rangle$, together with the total ionizing luminosity determined from the SLUG calculation, implicitly sets the density $n_{\rm H}$ that we use in the CLOUDY calculation. Note that, as long as $n_{\rm H}$ is much smaller than the critical density of any of the important collisionally excited lines, the exact value of $Q({\rm H}^0)$ and thus $n_{\rm H}$ changes only the absolute luminosities of the lines. The ratios of line luminosity to $Q({\rm H}^0)$, and thus the emission coefficients $\gamma^{\rm (M)}$, depend only $\langle \mathcal{U} \rangle$. Our grid of CLOUDY calculations uses $\log \langle \mathcal{U} \rangle = -3, -2.5$, and -2, which spans the plausible observed range.

After using CLOUDY to compute the properties of H II regions for each track set, age, and ionization parameter in our grid, we record the emission-weighted mean temperature

$\langle T\rangle = \int n_{\rm H}^2 T\,{\rm d}V / \int n_{\rm H}^2\,{\rm d}V$, and the ratio $L_{\rm line}/Q({\rm H}^0)$ for all non-hydrogen lines for which the ratio exceeds 10^{-20} erg photon^{-1} at any age; for comparison, this ratio is $\approx 10^{-12}$ erg photon^{-1} for bright lines such as H α. This cut typically leaves ∼80 significant lines. These data are stored in a tabular form. When a SLUG simulation is run, the user specifies an ionization parameter $\langle \mathcal{U}\rangle$. To compute nebular line emission, SLUG loads the tabulated data for the specified evolutionary tracks and $\langle \mathcal{U}\rangle$, and for each cluster or field star of known age interpolates on the grid of ages to determine $L_{\rm line}/Q({\rm H}^0)$ and thus $\gamma^{(\rm M)}$; field stars that are not being treated stochastically are handled using the values of $L_{\rm line}/Q({\rm H}^0)$ tabulated for continuous star formation. At the user's option, this procedure can also be used to obtain a temperature $\langle T\rangle$, which in turn can be used in all the calculations of H and He continuum emission, and H recombination line emission.

There is a final technical subtlety in SLUG's treatment of nebular emission. The wavelength grid found in most broad-band stellar atmosphere libraries, including those used by SLUG, is too coarse to represent a nebular emission line. This leads to potential problems with the representation of such lines, and the calculation of photometry in narrow-band filters targeting them (e.g. H α filters). To handle this issue, SLUG computes nebular spectra on a non-uniform grid in which extra wavelength resolution is added around the centre of each line. The extra grid points make it possible to resolve the shape of the line (which we compute by adopting a Gaussian line shape with a fiducial width of 20 km s^{-1}), at least marginally. The grid resolution is high enough so that it is possible to compute numerical integrals on the spectrum and recover the correct bolometric luminosity of each line to high accuracy, so that photometric outputs including the nebular contribution can be computed correctly.

B3 Dust extinction

SLUG parametrizes dust extinction via the V-band extinction, A_V. As noted in the main text, A_V can either be a constant value, or can be specified as a PDF as described in Appendix A1. In the latter case, every star cluster has its own extinction, so a range of extinctions are present. Once a value of A_V is specified, SLUG computes the wavelength-dependent extinction from a user-specified extinction law. The extinction curves that ship with the current version of the code are as follows:

(i) a Milky Way extinction curve, consisting of optical and UV extinctions taken from Fitzpatrick (1999), and IR extinctions taken from Landini et al. (1984), with the two parts combined by Calzetti (private communication);

(ii) a Large Magellanic Cloud extinction curve, taken from the same source as the Milky Way curve;

(iii) a Small Magellanic Cloud extinction curve, taken from Bouchet et al. (1985);

(iv) a 'starburst' extinction curve, taken from Calzetti et al. (2000).

APPENDIX C: SOFTWARE NOTES

Full details regarding the code implementation are included in the SLUG documentation, but we include in this appendix some details that are of general interest. First, SLUG, CLOUDY_SLUG, and various related software packages are fully parallelized for multicore environments. Secondly, the SLUG package includes a PYTHON helper library, SLUGPY, that is capable of reading and manipulating SLUG outputs, and which integrates with CLOUDY_SLUG, SFR_SLUG, and CLUSTER_SLUG. In addition to more mundane data processing tasks, the library supports ancillary computations such as convolving spectra with additional filters and converting data between photometric systems. The SLUGPY library is also fully integrated with all the tools described below. FITS file handling capabilities in SLUGPY are provided through ASTROPY (Astropy Collaboration et al. 2013). Thirdly, the code is highly modular so that it is easy to add additional options. Extinction curves and photometric filters are specified using extremely simple text file formats, so adding additional filters or extinction curves is simply a matter of placing additional text files in the relevant directories. Similarly, probability distributions describing IMFs, CMFs, SFHs, etc., are also specified via simple text files, so that additional choices can be added without needing to touch the source code in any way.

The numerical implementation used in BAYESPHOT requires particular attention, since having a fast implementation is critical for the code's utility. Since we have written the joint and marginal posterior probability distributions of the physical variables in terms of kernel density estimates (equations 21 and 24), we can perform numerical evaluation using standard fast methods. In BAYESPHOT, numerical evaluation proceeds in a number of steps. After reading in the library of simulations, we first compute the weights from the user-specified prior probability distributions and sampling densities (equation 18). We then store the sample points and weights in a k-dimensional (KD) tree structure. The bandwidth we choose for the kernel density estimation must be chosen appropriately for the input library of models, and for the underlying distribution they are modelling. There is no completely general procedure for making a 'good' choice for the bandwidth, so bandwidth selection is generally best done by hand.

Once the bandwidth has been chosen, we can evaluate the joint and marginal posterior PDFs to any desired level of accuracy by using the KD tree structure to avoid examining parts of the simulation library that are not relevant for any particular set of observations. As a result, once the tree has been constructed, the formal order of the algorithm for evaluating either the joint or marginal PDF using a library of $N_{\rm lib}$ simulations is only $\log N_{\rm lib}$, and in practice evaluations of the marginal PDF over relatively fine grids of points can be accomplished in well under a second on a single CPU, even for 5-band photometry and libraries of many millions of simulations. In addition to evaluating the joint or marginal PDF directly on a grid of sample points, if we are interested in the joint PDF of a relatively large number of physical variables, it may be desirable to use a Markov Chain Monte Carlo (MCMC) method to explore the shape of the posterior PDF. BAYESPHOT includes an interface to the MCMC code EMCEE (Foreman-Mackey et al. 2013), allowing transparent use of an MCMC technique as well as direct evaluation on a grid. However, if we are interested in the marginal PDFs only of one or two variables at time (for example the marginal PDF of star cluster mass or star cluster age, or their joint distribution), it is almost always faster to use equation (24) to evaluate this directly than to resort to MCMC. The ability to generate marginal posterior PDFs directly represents a significant advantage to our method, since this is often the quantity of greatest interest.

This paper has been typeset from a TEX/LATEX file prepared by the author.