

BWD AGWB

Generated by Doxygen 1.10.0

1 Namespace Index	1
1.1 Package List	1
2 Class Index	3
2.1 Class List	3
3 File Index	5
3.1 File List	5
4 Namespace Documentation	7
4.1 add_birth Namespace Reference	7
4.1.1 Function Documentation	7
4.1.1.1 add_birth()	7
4.2 add_bulk Namespace Reference	8
4.2.1 Function Documentation	8
4.2.1.1 add_bulk()	8
4.3 add_merge Namespace Reference	8
4.3.1 Function Documentation	8
4.3.1.1 add_merge()	8
4.4 auxiliary Namespace Reference	9
4.4.1 Function Documentation	10
4.4.1.1 calc_parabola_vertex()	10
4.4.1.2 determine_upper_freq()	11
4.4.1.3 get_bin_factors()	11
4.4.1.4 get_width_z_shell_from_z()	12
4.4.1.5 make_Omega_plot_unnorm()	12
4.4.1.6 Omega()	12
4.4.1.7 parabola()	13
4.4.1.8 tau_syst()	13
4.4.2 Variable Documentation	14
4.4.2.1 s_in_Myr	14
4.5 GWB Namespace Reference	14
4.5.1 Function Documentation	14
4.5.1.1 main()	14
4.5.2 Variable Documentation	14
4.5.2.1 action	14
4.5.2.2 category	14
4.5.2.3 fontsize	15
4.5.2.4 labelsiz	15
4.5.2.5 s_in_Myr	15
4.5.2.6 size	15
4.5.2.7 titlesize	15
4.6 num_syst Namespace Reference	15

4.6.1 Function Documentation	16
4.6.1.1 determine_upper_freq()	16
4.6.1.2 get_width_z_shell_from_z()	16
4.6.1.3 get_z_fast()	16
4.6.1.4 main()	16
4.6.1.5 num_merge_bins()	17
4.6.1.6 representative_SFH()	17
4.6.1.7 SFH()	17
4.6.1.8 tau_syst()	17
4.6.2 Variable Documentation	17
4.6.2.1 fontsize	17
4.6.2.2 interp_age	17
4.6.2.3 interp_z	18
4.6.2.4 labelsizes	18
4.6.2.5 s_in_Myr	18
4.6.2.6 size	18
4.6.2.7 titlesize	18
4.6.2.8 z_at_val_data	18
4.7 RedshiftInterpolator Namespace Reference	18
4.7.1 Detailed Description	18
4.8 SFH Namespace Reference	19
4.8.1 Function Documentation	19
4.8.1.1 representative_SFH()	19
4.8.1.2 SFH2()	19
4.8.1.3 SFH3()	21
4.8.1.4 SFH4()	21
4.8.1.5 SFH_MD()	21
4.9 SimModel Namespace Reference	22
4.9.1 Detailed Description	22
5 Class Documentation	23
5.1 RedshiftInterpolator.RedshiftInterpolator Class Reference	23
5.1.1 Detailed Description	23
5.1.2 Constructor & Destructor Documentation	23
5.1.2.1 __init__()	23
5.1.3 Member Function Documentation	24
5.1.3.1 get_z_fast()	24
5.1.4 Member Data Documentation	24
5.1.4.1 interp_age	24
5.1.4.2 interp_z	24
5.2 num_syst.sim_model Class Reference	24
5.2.1 Detailed Description	25

5.2.2 Constructor & Destructor Documentation	25
5.2.2.1 <code>__init__()</code>	25
5.2.3 Member Function Documentation	25
5.2.3.1 <code>calculate_cosmology()</code>	25
5.2.3.2 <code>calculate_f_bins()</code>	26
5.2.3.3 <code>calculate_z_bins()</code>	26
5.2.4 Member Data Documentation	26
5.2.4.1 <code>ages</code>	26
5.2.4.2 <code>f_bins</code>	26
5.2.4.3 <code>f_plot</code>	26
5.2.4.4 <code>log_f_high</code>	26
5.2.4.5 <code>log_f_low</code>	26
5.2.4.6 <code>max_z</code>	26
5.2.4.7 <code>N</code>	26
5.2.4.8 <code>N_z</code>	27
5.2.4.9 <code>SFH_num</code>	27
5.2.4.10 <code>z_bins</code>	27
5.2.4.11 <code>z_list</code>	27
5.2.4.12 <code>z_time_since_max_z</code>	27
5.2.4.13 <code>z_widths</code>	27
5.3 SimModel.SimModel Class Reference	27
5.3.1 Detailed Description	28
5.3.2 Constructor & Destructor Documentation	29
5.3.2.1 <code>__init__()</code>	29
5.3.3 Member Function Documentation	29
5.3.3.1 <code>calculate_cosmology_from_T()</code>	29
5.3.3.2 <code>calculate_cosmology_from_z()</code>	29
5.3.3.3 <code>calculate_f_bins()</code>	30
5.3.3.4 <code>calculate_T_bins()</code>	30
5.3.3.5 <code>calculate_z_bins()</code>	30
5.3.3.6 <code>set_mode()</code>	30
5.3.4 Member Data Documentation	30
5.3.4.1 <code>ages</code>	30
5.3.4.2 <code>DEBUG</code>	31
5.3.4.3 <code>dT</code>	31
5.3.4.4 <code>f_bin_factors</code>	31
5.3.4.5 <code>f_bins</code>	31
5.3.4.6 <code>f_plot</code>	31
5.3.4.7 <code>INTEG_MODE</code>	31
5.3.4.8 <code>light_speed</code>	31
5.3.4.9 <code>log_f_high</code>	31
5.3.4.10 <code>log_f_low</code>	31

5.3.4.11 max_z	32
5.3.4.12 N_freq	32
5.3.4.13 N_int	32
5.3.4.14 SAVE_FIG	32
5.3.4.15 SFH_num	32
5.3.4.16 T0	32
5.3.4.17 T_bins	32
5.3.4.18 T_list	32
5.3.4.19 T_range	32
5.3.4.20 TEST_FOR_ONE	32
5.3.4.21 z_bins	33
5.3.4.22 z_list	33
5.3.4.23 z_time_since_max_z	33
5.3.4.24 z_widths	33
6 File Documentation	35
6.1 /home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/GWB.py File Reference	35
6.1.1 Detailed Description	35
6.2 /home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/add↔_birth.py File Reference	36
6.2.1 Detailed Description	36
6.3 /home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/add↔_bulk.py File Reference	36
6.3.1 Detailed Description	37
6.4 /home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/add↔_merge.py File Reference	37
6.4.1 Detailed Description	37
6.5 /home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/auxiliary.py File Reference	37
6.5.1 Detailed Description	38
6.6 /home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/↔_RedshiftInterpolator.py File Reference	38
6.7 /home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/↔_SFH.py File Reference	39
6.7.1 Detailed Description	39
6.8 /home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/Sim↔_Model.py File Reference	40
6.9 /home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/num_syst.py File Reference	40
Index	41

Chapter 1

Namespace Index

1.1 Package List

Here are the packages with brief descriptions (if available):

add_birth	7
add_bulk	8
add_merge	8
auxiliary	9
GWB	14
num_syst	15
RedshiftInterpolator	
This module contains the class RedshiftInterpolator	18
SFH	19
SimModel	
This module contains the class SimModel	22

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

RedshiftInterpolator.RedshiftInterpolator	
This class is used to quickly determine the redshift at a given age of the Universe	23
num_syst.sim_model	
MODEL CLASS #####	24
SimModel.SimModel	
! This class contains information about the run that needs to be shared over the different sub-routines	27

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/GWB.py	
This program calculates the GWB based on the method described in my thesis, using uniform redshift bins	35
/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/num_syst.py	40
/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/add_birth.py	
This file contains a routine that adds the contribution of the 'birth bins' to the bulk GWB	36
/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/add_bulk.py	
This file contains a routine that calculates the majority of the GWB , what is referred to in my thesis as the 'generic case'	36
/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/add_merge.py	
This file contains a routine that adds the contribution of the 'merger bins' due to Kepler max to the bulk+birth GWB	37
/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/auxiliary.py	
This module contains auxiliary functions that are used in the main code	37
/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/RedshiftInterpolator.py	38
/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/SFH.py	
This file contains the functions to determine the star formation rate	39
/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/SimModel.py	40

Chapter 4

Namespace Documentation

4.1 `add_birth` Namespace Reference

Functions

- None `add_birth` (sm.SimModel model, ri.RedshiftInterpolator z_interp, pd.DataFrame data, str tag)

This routine adds the contribution of the 'birth bins' to the bulk [GWB](#).

4.1.1 Function Documentation

4.1.1.1 `add_birth()`

```
None add_birth.add_birth (
    sm.SimModel model,
    ri.RedshiftInterpolator z_interp,
    pd.DataFrame data,
    str tag )
```

This routine adds the contribution of the 'birth bins' to the bulk [GWB](#).

Parameters

<i>model</i>	instance of SimModel , containing the necessary information for the run.
<i>z_interp</i>	instance of RedshiftInterpolator , used in the SFH calculations.
<i>data</i>	dataframe containing the binary population data.
<i>tag</i>	tag to add to the output files.

Returns

Saves a dataframe with all the essential information.

4.2 `add_bulk` Namespace Reference

Functions

- None `add_bulk` (sm.SimModel model, ri.RedshiftInterpolator z_interp, pd.DataFrame data, str tag)

This routine calculates the majority of the [GWB](#), what is referred to in my thesis as the 'generic case'.

4.2.1 Function Documentation

4.2.1.1 `add_bulk()`

```
None add_bulk.add_bulk (
    sm.SimModel model,
    ri.RedshiftInterpolator z_interp,
    pd.DataFrame data,
    str tag )
```

This routine calculates the majority of the [GWB](#), what is referred to in my thesis as the 'generic case'.

Parameters

<i>model</i>	instance of SimModel , containing the necessary information for the run.
<i>z_interp</i>	instance of RedshiftInterpolator , used in the SFH calculations.
<i>data</i>	dataframe containing the binary population data.
<i>tag</i>	tag to add to the output files.

Returns

Saves a dataframe with all the essential information.

4.3 `add_merge` Namespace Reference

Functions

- None `add_merge` (sm.SimModel model, ri.RedshiftInterpolator z_interp, pd.DataFrame data, str tag)

This routine adds the contribution of the 'merger bins' due to Kepler max to the bulk+birth [GWB](#).

4.3.1 Function Documentation

4.3.1.1 `add_merge()`

```
None add_merge.add_merge (
    sm.SimModel model,
```

```

    ri.RedshiftInterpolator z_interp,
    pd.DataFrame data,
    str tag )

```

This routine adds the contribution of the 'merger bins' due to Kepler max to the bulk+birth [GWB](#).

Parameters

<i>model</i>	instance of SimModel , containing the necessary information for the run.
<i>z_interp</i>	instance of RedshiftInterpolator , used in the SFH calculations.
<i>data</i>	dataframe containing the binary population data.
<i>tag</i>	tag to add to the output files.

Returns

Saves a dataframe with all the essential information.

4.4 auxiliary Namespace Reference

Functions

- tuple [calc_parabola_vertex](#) (float x1, float y1, float x2, float y2, float x3, float y3)
Calculate the coefficients of a parabola given three points.
- float [parabola](#) (float x, float a, float b, float c)
Calculate the value of a parabola given the coefficients.
- np.array [get_bin_factors](#) (np.array freqs, np.array bins)
Determine bin factors that often recur in the calculation to store them.
- np.array [get_width_z_shell_from_z](#) (np.array z_vals)
Returns the widths of the redshift shells in Mpc.
- np.array [Omega](#) (float Omega_ref, float f_ref, np.array freq)
Create a $f^{2/3}$ spectrum line.
- None [make_Omega_plot_unnorm](#) (np.array f, np.array Omega_sim, bool save=False, str save_name="void", bool show=False)
Make a plot showing Omega for BWD.
- float [tau_syst](#) (float f_0, float f_1, float K, float [s_in_Myr](#))
Calculates tau, the time it takes a binary with K to evolve from f_0 to f_1 (GW frequencies).
- float [determine_upper_freq](#) (float nu_low, float evolve_time, float K, bool DEBUG=False)
Determines upper ORBITAL frequency for a binary with K, starting from nu_0, evolving over evolve_time.

Variables

- tuple [s_in_Myr](#) = (u.Myr).to(u.s)

4.4.1 Function Documentation

4.4.1.1 `calc_parabola_vertex()`

```
tuple auxiliary.calc_parabola_vertex (  
    float x1,  
    float y1,  
    float x2,  
    float y2,  
    float x3,  
    float y3 )
```

Calculate the coefficients of a parabola given three points.

Parameters

<i>x1,y1</i>	x and y coordinates of the first point.
<i>x2,y2</i>	x and y coordinates of the second point.
<i>x3,y3</i>	x and y coordinates of the third point.

Returns

A, B, C: coefficients of the parabola.

4.4.1.2 determine_upper_freq()

```
float auxiliary.determine_upper_freq (
    float nu_low,
    float evolve_time,
    float K,
    bool DEBUG = False )
```

Determines upper ORBITAL frequency for a binary with K, starting from nu_0, evolving over evolve_time.

Parameters

<i>nu_low</i>	initial orbital frequency.
<i>evolve_time</i>	time it takes to evolve in Myr.
<i>K</i>	constant depending on the binary.

Returns

nu_upp: upper orbital frequency.

4.4.1.3 get_bin_factors()

```
np.array auxiliary.get_bin_factors (
    np.array freqs,
    np.array bins )
```

Determine bin factors that often recur in the calculation to store them.

Parameters

<i>freqs</i>	central frequencies.
<i>bins</i>	frequency bin edges.

Returns

factors: factors to multiply the contributions with.

4.4.1.4 `get_width_z_shell_from_z()`

```
np.array auxiliary.get_width_z_shell_from_z (
    np.array z_vals )
```

Returns the widths of the redshift shells in Mpc.

Parameters

<i>z_vals</i>	redshift values.
---------------	------------------

Returns

shells: shell widths in Mpc.

4.4.1.5 `make_Omega_plot_unnorm()`

```
None auxiliary.make_Omega_plot_unnorm (
    np.array f,
    np.array Omega_sim,
    bool save = False,
    str save_name = "void",
    bool show = False )
```

Make a plot showing Omega for BWD.

Parameters

<i>f</i>	frequency array.
<i>Omega_sim</i>	Omega array.
<i>save</i>	save the figure.
<i>save_name</i>	name of the saved figure.
<i>show</i>	show the figure.

4.4.1.6 `Omega()`

```
np.array auxiliary.Omega (
    float Omega_ref,
    float f_ref,
    np.array freq )
```

Create a $f^{2/3}$ spectrum line.

Parameters

<i>Omega_ref</i>	reference Omega value.
<i>f_ref</i>	reference frequency.
<i>freq</i>	frequency array.

Returns

Omega: Omega array.

4.4.1.7 parabola()

```
float auxiliary.parabola (
    float x,
    float a,
    float b,
    float c )
```

Calculate the value of a parabola given the coefficients.

Parameters

x	x value.
a,b,c	coefficients of the parabola.

Returns

y: y value.

4.4.1.8 tau_syst()

```
float auxiliary.tau_syst (
    float  $f_0$ ,
    float  $f_1$ ,
    float  $K$ ,
    float  $s_{in\_Myr}$  )
```

Calculates tau, the time it takes a binary with K to evolve from f_0 to f_1 (GW frequencies).

Parameters

f_0	initial frequency.
f_1	final frequency.
K	constant depending on the binary.

Returns

tau: time in Myr.

4.4.2 Variable Documentation

4.4.2.1 s_in_Myr

```
tuple auxiliary.s_in_Myr = (u.Myr).to(u.s)
```

4.5 GWB Namespace Reference

Functions

- [main\(\)](#)
Main function.

Variables

- [size](#)
- [titlesize](#)
- [labelsize](#)
- [fontsize](#)
- [action](#)
- [category](#)
- tuple [s_in_Myr](#) = (u.Myr).to(u.s)

4.5.1 Function Documentation

4.5.1.1 main()

```
GWB.main ( )
```

Main function.

The main functions sets the details of the simulation and runs the three main parts of the program.

4.5.2 Variable Documentation

4.5.2.1 action

```
GWB.action
```

4.5.2.2 category

```
GWB.category
```

4.5.2.3 fontsize

GWB.fontsize

4.5.2.4 labelsiz

GWB.labelsiz

4.5.2.5 s_in_Myr

tuple GWB.s_in_Myr = (u.Myr).to(u.s)

4.5.2.6 size

GWB.size

4.5.2.7 titlesiz

GWB.titlesiz

4.6 num_syst Namespace Reference

Classes

- class [sim_model](#)
MODEL CLASS #####.

Functions

- [get_width_z_shell_from_z](#) (z_vals)
AUXILIARY FUNCTIONS #####.
- [SFH](#) (z)
Star formation history from [Madau, Dickinson 2014].
- [tau_syst](#) (f_0, f_1, K)
Calculates tau, the time it takes a binary with K to evolve from f_0 to f_1 (GW frequencies).
- [representative_SFH](#) (age, Delta_t, SFH_num, max_z)
Looks for a representative value of the SFH given the age of the system, and an additional time delay in reaching the bin.
- [get_z_fast](#) (age)
- [determine_upper_freq](#) (nu_low, evolve_time, K)
Determines upper ORBITAL frequency for a binary with K, starting from nu_0, evolving over evolve_time.
- [num_merge_bins](#) (model1, model2, data, tag)
- [main](#) ()
ACTUAL MAIN FUNCTION #####.

Variables

- [size](#)
- [titlesize](#)
- [labelsizes](#)
- [fontsize](#)
- tuple [s_in_Myr](#) = (u.Myr).to(u.s)
- [z_at_val_data](#) = pd.read_csv("../Data/z_at_age.txt", names=["age", "z"], header=1)
LOAD Z_AT_VALUE FILE #####.
- [interp_age](#)
- [interp_z](#)

4.6.1 Function Documentation

4.6.1.1 determine_upper_freq()

```
num_syst.determine_upper_freq (
    nu_low,
    evolve_time,
    K )
```

Determines upper ORBITAL frequency for a binary with K, starting from nu_0, evolving over evolve_time.

Takes evolve_time in Myr, so needs to be converted.

4.6.1.2 get_width_z_shell_from_z()

```
num_syst.get_width_z_shell_from_z (
    z_vals )
```

AUXILIARY FUNCTIONS #####.

Returns the widths of the z_shells in Mpc.

4.6.1.3 get_z_fast()

```
num_syst.get_z_fast (
    age )
```

4.6.1.4 main()

```
num_syst.main ( )
```

ACTUAL MAIN FUNCTION #####.

The actual main function. Combines the three different components

4.6.1.5 num_merge_bins()

```
num_syst.num_merge_bins (
    model1,
    model2,
    data,
    tag )
```

4.6.1.6 representative_SFH()

```
num_syst.representative_SFH (
    age,
    Delta_t,
    SFH_num,
    max_z )
```

Looks for a representative value of the [SFH](#) given the age of the system, and an additional time delay in reaching the bin.

age and Delta_t should be given in Myr.

4.6.1.7 SFH()

```
num_syst.SFH (
    z )
```

Star formation history from [Madau, Dickinson 2014].

Units: solar mass / yr / Mpc³

4.6.1.8 tau_syst()

```
num_syst.tau_syst (
    f_0,
    f_1,
    K )
```

Calculates tau, the time it takes a binary with K to evolve from f_0 to f_1 (GW frequencies).

Returns tau in Myr.

4.6.2 Variable Documentation

4.6.2.1 fontsize

```
num_syst.fontsize
```

4.6.2.2 interp_age

```
num_syst.interp_age
```

4.6.2.3 interp_z

```
num_syst.interp_z
```

4.6.2.4 labelsiz

```
num_syst.labelsiz
```

4.6.2.5 s_in_Myr

```
tuple num_syst.s_in_Myr = (u.Myr).to(u.s)
```

4.6.2.6 siz

```
num_syst.siz
```

4.6.2.7 titlesiz

```
num_syst.titlesiz
```

4.6.2.8 z_at_val_data

```
num_syst.z_at_val_data = pd.read_csv("../Data/z_at_age.txt", names=["age", "z"], header=1)
```

```
LOAD Z_AT_VALUE FILE #####.
```

4.7 RedshiftInterpolator Namespace Reference

This module contains the class [RedshiftInterpolator](#).

Classes

- class [RedshiftInterpolator](#)

This class is used to quickly determine the redshift at a given age of the Universe.

4.7.1 Detailed Description

This module contains the class [RedshiftInterpolator](#).

The class [RedshiftInterpolator](#) is used to quickly determine the redshift at a given age of the Universe.

Author

Seppe Staelens

Date

2024-07-24

4.8 SFH Namespace Reference

Functions

- [representative_SFH](#) (float age, [ri.RedshiftInterpolator](#) redshift_interpolator, float Delta_t=0., int SFH_num=1, float max_z=8.)
Determines an appropriate value for the star formation rate at a given age.
- float [SFH_MD](#) (float z)
Star formation history from [Madau, Dickinson 2014].
- float [SFH2](#) (float z)
Made up star formation history.
- float [SFH3](#) (float z)
Made up star formation history.
- float [SFH4](#) (float z)
Made up star formation history.

4.8.1 Function Documentation

4.8.1.1 representative_SFH()

```
SFH.representative_SFH (
    float age,
    ri.RedshiftInterpolator redshift_interpolator,
    float Delta_t = 0.,
    int SFH_num = 1,
    float max_z = 8. )
```

Determines an appropriate value for the star formation rate at a given age.

The function looks for a representative value of the star formation rate given the age of the system, and takes into account an optional additional time delay.

Parameters

<i>age</i>	age of the system in Myr.
<i>redshift_interpolator</i>	RedshiftInterpolator object that interpolates the redshift at a given age.
<i>Delta_t</i>	time delay due to formation of binary or time required to reach the correct frequency bin, in Myr.
<i>SFH_num</i>	which star formation history to select. 1: Madau & Dickinson 2014, 2-4: made up, 5: constant 0.01.
<i>max_z</i>	maximum redshift.

Returns

SFR: star formation rate. Units: solar mass / yr / Mpc³.

4.8.1.2 SFH2()

```
float SFH.SFH2 (
    float z )
```

Made up star formation history.

Parameters

z	redshift.
---	-----------

Returns

SFR: star formation rate. Units: solar mass / yr / Mpc³.

4.8.1.3 SFH3()

```
float SFH.SFH3 (  
    float z )
```

Made up star formation history.

Parameters

z	redshift.
---	-----------

Returns

SFR: star formation rate. Units: solar mass / yr / Mpc³.

4.8.1.4 SFH4()

```
float SFH.SFH4 (  
    float z )
```

Made up star formation history.

Parameters

z	redshift.
---	-----------

Returns

SFR: star formation rate. Units: solar mass / yr / Mpc³.

4.8.1.5 SFH_MD()

```
float SFH.SFH_MD (  
    float z )
```

Star formation history from [Madau, Dickinson 2014].

Parameters

z	redshift.
---	-----------

Returns

SFR: star formation rate. Units: solar mass / yr / Mpc³.

4.9 SimModel Namespace Reference

This module contains the class [SimModel](#).

Classes

- class [SimModel](#)

! This class contains information about the run that needs to be shared over the different subroutines.

4.9.1 Detailed Description

This module contains the class [SimModel](#).

The class [SimModel](#) contains information about the run that needs to be shared over the different subroutines.

Author

Seppe Staelens

Date

2024-07-24

Chapter 5

Class Documentation

5.1 RedshiftInterpolator.RedshiftInterpolator Class Reference

This class is used to quickly determine the redshift at a given age of the Universe.

Public Member Functions

- None `__init__` (self, str z_at_age_file)
Initializes the [RedshiftInterpolator](#) object.
- float `get_z_fast` (self, float age)
Quickly determine the redshift at a given age of the Universe.

Public Attributes

- `interp_age`
The age of the Universe at which the redshift is determined.
- `interp_z`
The redshift at the given age of the Universe.

5.1.1 Detailed Description

This class is used to quickly determine the redshift at a given age of the Universe.

5.1.2 Constructor & Destructor Documentation

5.1.2.1 `__init__()`

```
None RedshiftInterpolator.RedshiftInterpolator.__init__ (  
    self,  
    str z_at_age_file )
```

Initializes the [RedshiftInterpolator](#) object.

Parameters

<code>z_at_age_file</code>	file containing the redshift at a given age of the Universe.
----------------------------	--

5.1.3 Member Function Documentation

5.1.3.1 `get_z_fast()`

```
float RedshiftInterpolator.RedshiftInterpolator.get_z_fast (
    self,
    float age )
```

Quickly determine the redshift at a given age of the Universe.

Parameters

<code>age</code>	age of the Universe in Myr.
------------------	-----------------------------

Returns

redshift at the given age of the Universe.

5.1.4 Member Data Documentation

5.1.4.1 `interp_age`

```
RedshiftInterpolator.RedshiftInterpolator.interp_age
```

The age of the Universe at which the redshift is determined.

5.1.4.2 `interp_z`

```
RedshiftInterpolator.RedshiftInterpolator.interp_z
```

The redshift at the given age of the Universe.

The documentation for this class was generated from the following file:

- `/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/RedshiftInterpolator.py`

5.2 `num_syst.sim_model` Class Reference

MODEL CLASS #####.

Public Member Functions

- None `__init__` (self, `N`=50, `N_z`=20, `max_z`=8, `SFH_num`=1, `log_f_low`=-5, `log_f_high`=0)
- `calculate_f_bins` (self)
Calculates the f bins and the bin factors.
- `calculate_z_bins` (self)
Calculates the z bins.
- `calculate_cosmology` (self)

Public Attributes

- `N`
- `N_z`
- `max_z`
- `SFH_num`
- `log_f_low`
- `log_f_high`
- `f_plot`
- `f_bins`
- `z_list`
- `z_bins`
- `z_widths`
- `z_time_since_max_z`
- `ages`

5.2.1 Detailed Description

MODEL CLASS #####.

This class contains information about the run that needs to be shared over the different subroutines.

5.2.2 Constructor & Destructor Documentation

5.2.2.1 `__init__()`

```
None num_syst.sim_model.__init__ (
    self,
    N = 50,
    N_z = 20,
    max_z = 8,
    SFH_num = 1,
    log_f_low = -5,
    log_f_high = 0 )
```

5.2.3 Member Function Documentation

5.2.3.1 `calculate_cosmology()`

```
num_syst.sim_model.calculate_cosmology (
    self )
```

5.2.3.2 calculate_f_bins()

```
num_syst.sim_model.calculate_f_bins (
    self )
```

Calculates the f bins and the bin factors.

5.2.3.3 calculate_z_bins()

```
num_syst.sim_model.calculate_z_bins (
    self )
```

Calculates the z bins.

5.2.4 Member Data Documentation

5.2.4.1 ages

```
num_syst.sim_model.ages
```

5.2.4.2 f_bins

```
num_syst.sim_model.f_bins
```

5.2.4.3 f_plot

```
num_syst.sim_model.f_plot
```

5.2.4.4 log_f_high

```
num_syst.sim_model.log_f_high
```

5.2.4.5 log_f_low

```
num_syst.sim_model.log_f_low
```

5.2.4.6 max_z

```
num_syst.sim_model.max_z
```

5.2.4.7 N

```
num_syst.sim_model.N
```


5.2.4.8 N_z

```
num_syst.sim_model.N_z
```

5.2.4.9 SFH_num

```
num_syst.sim_model.SFH_num
```

5.2.4.10 z_bins

```
num_syst.sim_model.z_bins
```

5.2.4.11 z_list

```
num_syst.sim_model.z_list
```

5.2.4.12 z_time_since_max_z

```
num_syst.sim_model.z_time_since_max_z
```

5.2.4.13 z_widths

```
num_syst.sim_model.z_widths
```

The documentation for this class was generated from the following file:

- /home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/[num_syst.py](#)

5.3 SimModel.SimModel Class Reference

! This class contains information about the run that needs to be shared over the different subroutines.

Public Member Functions

- None `__init__` (self, str `INTEG_MODE`, int `z_interp`, int `N_freq`=50, int `N_int`=20, float `max_z`=8, int `SFH_num`=1, float `log_f_low`=-5, float `log_f_high`=0)
Initializes the `SimModel` object.
- None `calculate_f_bins` (self)
Calculates the `f` bins and the bin factors.
- None `calculate_z_bins` (self)
Calculates the `z` bins.
- None `calculate_T_bins` (self)
Calculates the `T` bins.
- None `calculate_cosmology_from_z` (self)
Calculations depending on the cosmology, starting from redshift bins.
- None `calculate_cosmology_from_T` (self, `ri.RedshiftInterpolator z_interpolator`)
Calculations depending on the cosmology, starting from cosmic time bins.
- None `set_mode` (self, bool `SAVE_FIG`, bool `DEBUG`, bool `TEST_FOR_ONE`)
Sets the mode of the simulation.

Public Attributes

- [N_freq](#)
- [N_int](#)
- [max_z](#)
- [SFH_num](#)
- [log_f_low](#)
- [log_f_high](#)
- [INTEG_MODE](#)
- [f_plot](#)

The frequencies at which we will plot.
- [f_bins](#)

The frequency bins.
- [f_bin_factors](#)

The frequency bin factors that appear in the calculation.
- [z_list](#)

The central values of the redshift bins.
- [z_bins](#)

The redshift bins.
- [T0](#)
- [T_range](#)
- [T_list](#)
- [T_bins](#)
- [dT](#)
- [z_widths](#)

The width of the redshift bins in Mpc.
- [z_time_since_max_z](#)

The time since the maximum redshift
- [ages](#)

The age of the universe at each redshift.
- [SAVE_FIG](#)
- [DEBUG](#)
- [TEST_FOR_ONE](#)

Static Public Attributes

- float [light_speed](#) = 0.30660139

The speed of light in units of Mpc/Myr.

5.3.1 Detailed Description

! This class contains information about the run that needs to be shared over the different subroutines.

5.3.2 Constructor & Destructor Documentation

5.3.2.1 __init__()

```
None SimModel.SimModel.__init__ (
    self,
    str INTEG_MODE,
    int z_interp,
    int N_freq = 50,
    int N_int = 20,
    float max_z = 8,
    int SFH_num = 1,
    float log_f_low = -5,
    float log_f_high = 0 )
```

Initializes the [SimModel](#) object.

Parameters

<i>N_freq</i>	number of frequency bins.
<i>N_int</i>	number of integration bins (z or T).
<i>max_z</i>	maximum redshift.
<i>SFH_num</i>	which star formation history to select. 1: Madau & Dickinson 2014, 2-4: made up, 5: constant 0.01.
<i>log_f_low</i>	lower bound of the frequency bins in log10 space.
<i>log_f_high</i>	upper bound of the frequency bins in log10 space.

Returns

instance of [SimModel](#), with frequency and redshift bins calculated, and cosmology set.

5.3.3 Member Function Documentation

5.3.3.1 calculate_cosmology_from_T()

```
None SimModel.SimModel.calculate_cosmology_from_T (
    self,
    ri.RedshiftInterpolator z_interpolator )
```

Calculations depending on the cosmology, starting from cosmic time bins.

Calculates the redshifts, the time since the maximum redshift, and the ages of the universe at each time.

5.3.3.2 calculate_cosmology_from_z()

```
None SimModel.SimModel.calculate_cosmology_from_z (
    self )
```

Calculations depending on the cosmology, starting from redshift bins.

Sets the widths of the z bins and the time since max z, as well as the age of the universe at each redshift.

5.3.3.3 calculate_f_bins()

```
None SimModel.SimModel.calculate_f_bins (
    self )
```

Calculates the f bins and the bin factors.

5.3.3.4 calculate_T_bins()

```
None SimModel.SimModel.calculate_T_bins (
    self )
```

Calculates the T bins.

5.3.3.5 calculate_z_bins()

```
None SimModel.SimModel.calculate_z_bins (
    self )
```

Calculates the z bins.

5.3.3.6 set_mode()

```
None SimModel.SimModel.set_mode (
    self,
    bool SAVE_FIG,
    bool DEBUG,
    bool TEST_FOR_ONE )
```

Sets the mode of the simulation.

Parameters

<i>SAVE_FIG</i>	whether to save the figures.
<i>DEBUG</i>	whether to print more output.
<i>TEST_FOR_ONE</i>	whether to test for only one system.
<i>INT_MODE</i>	whether to integrate over redshift or time.

5.3.4 Member Data Documentation

5.3.4.1 ages

```
SimModel.SimModel.ages
```

The age of the universe at each redshift.

5.3.4.2 DEBUG

```
SimModel.SimModel.DEBUG
```

5.3.4.3 dT

```
SimModel.SimModel.dT
```

5.3.4.4 f_bin_factors

```
SimModel.SimModel.f_bin_factors
```

The frequency bin factors that appear in the calculation.

5.3.4.5 f_bins

```
SimModel.SimModel.f_bins
```

The frequency bins.

5.3.4.6 f_plot

```
SimModel.SimModel.f_plot
```

The frequencies at which we will plot.

5.3.4.7 INTEG_MODE

```
SimModel.SimModel.INTEG_MODE
```

5.3.4.8 light_speed

```
float SimModel.SimModel.light_speed = 0.30660139 [static]
```

The speed of light in units of Mpc/Myr.

5.3.4.9 log_f_high

```
SimModel.SimModel.log_f_high
```

5.3.4.10 log_f_low

```
SimModel.SimModel.log_f_low
```

5.3.4.11 max_z

```
SimModel.SimModel.max_z
```

5.3.4.12 N_freq

```
SimModel.SimModel.N_freq
```

5.3.4.13 N_int

```
SimModel.SimModel.N_int
```

5.3.4.14 SAVE_FIG

```
SimModel.SimModel.SAVE_FIG
```

5.3.4.15 SFH_num

```
SimModel.SimModel.SFH_num
```

5.3.4.16 T0

```
SimModel.SimModel.T0
```

5.3.4.17 T_bins

```
SimModel.SimModel.T_bins
```

5.3.4.18 T_list

```
SimModel.SimModel.T_list
```

5.3.4.19 T_range

```
SimModel.SimModel.T_range
```

5.3.4.20 TEST_FOR_ONE

```
SimModel.SimModel.TEST_FOR_ONE
```

5.3.4.21 z_bins

```
SimModel.SimModel.z_bins
```

The redshift bins.

5.3.4.22 z_list

```
SimModel.SimModel.z_list
```

The central values of the redshift bins.

5.3.4.23 z_time_since_max_z

```
SimModel.SimModel.z_time_since_max_z
```

The time since the maximum redshift

5.3.4.24 z_widths

```
SimModel.SimModel.z_widths
```

The width of the redshift bins in Mpc.

The documentation for this class was generated from the following file:

- /home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/[SimModel.py](#)

Chapter 6

File Documentation

6.1 /home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_↔ Dwarf_AGWB/src/GWB.py File Reference

This program calculates the [GWB](#) based on the method described in my thesis, using uniform redshift bins.

Namespaces

- namespace [GWB](#)

Functions

- [GWB.main](#) ()
Main function.

Variables

- [GWB.size](#)
- [GWB.titlesize](#)
- [GWB.labelsize](#)
- [GWB.fontsize](#)
- [GWB.action](#)
- [GWB.category](#)
- tuple [GWB.s_in_Myr](#) = (u.Myr).to(u.s)

6.1.1 Detailed Description

This program calculates the [GWB](#) based on the method described in my thesis, using uniform redshift bins.

Date

2024-07-26

The program calculates the [GWB](#) based on the method described in my thesis, using uniform redshift bins. It is divided into three main parts: the bulk part, the birth part, and the merger part. The bulk part calculates the majority of the [GWB](#), what is referred to in my thesis as the 'generic case'. The birth part adds the contribution of the 'birth bins' to the bulk [GWB](#). The merger part adds the contribution of the 'merger bins' due to Kepler max to the bulk [GWB](#). The program saves a dataframe with all the essential information.

Author

Seppe Staelens

6.2 `/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/add_birth.py` File Reference

This file contains a routine that adds the contribution of the 'birth bins' to the bulk [GWB](#).

Namespaces

- namespace [add_birth](#)

Functions

- None [add_birth.add_birth](#) (sm.SimModel model, ri.RedshiftInterpolator z_interp, pd.DataFrame data, str tag)
This routine adds the contribution of the 'birth bins' to the bulk [GWB](#).

6.2.1 Detailed Description

This file contains a routine that adds the contribution of the 'birth bins' to the bulk [GWB](#).

Author

Seppe Staelens

Date

2024-07-24

6.3 `/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/add_bulk.py` File Reference

This file contains a routine that calculates the majority of the [GWB](#), what is referred to in my thesis as the 'generic case'.

Namespaces

- namespace [add_bulk](#)

Functions

- None [add_bulk.add_bulk](#) (sm.SimModel model, ri.RedshiftInterpolator z_interp, pd.DataFrame data, str tag)
This routine calculates the majority of the [GWB](#), what is referred to in my thesis as the 'generic case'.

6.3.1 Detailed Description

This file contains a routine that calculates the majority of the [GWB](#), what is referred to in my thesis as the 'generic case'.

Author

Seppe Staelens

Date

2024-07-24

6.4 /home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/add_merge.py File Reference

This file contains a routine that adds the contribution of the 'merger bins' due to Kepler max to the bulk+birth [GWB](#).

Namespaces

- namespace [add_merge](#)

Functions

- None [add_merge.add_merge](#) (sm.SimModel model, ri.RedshiftInterpolator z_interp, pd.DataFrame data, str tag)

This routine adds the contribution of the 'merger bins' due to Kepler max to the bulk+birth [GWB](#).

6.4.1 Detailed Description

This file contains a routine that adds the contribution of the 'merger bins' due to Kepler max to the bulk+birth [GWB](#).

Author

Seppe Staelens

Date

2024-07-24

6.5 /home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/auxiliary.py File Reference

This module contains auxiliary functions that are used in the main code.

Namespaces

- namespace [auxiliary](#)

Functions

- tuple [auxiliary.calc_parabola_vertex](#) (float x1, float y1, float x2, float y2, float x3, float y3)
Calculate the coefficients of a parabola given three points.
- float [auxiliary.parabola](#) (float x, float a, float b, float c)
Calculate the value of a parabola given the coefficients.
- np.array [auxiliary.get_bin_factors](#) (np.array freqs, np.array bins)
Determine bin factors that often recur in the calculation to store them.
- np.array [auxiliary.get_width_z_shell_from_z](#) (np.array z_vals)
Returns the widths of the redshift shells in Mpc.
- np.array [auxiliary.Omega](#) (float Omega_ref, float f_ref, np.array freq)
Create a $f^{2/3}$ spectrum line.
- None [auxiliary.make_Omega_plot_unnorm](#) (np.array f, np.array Omega_sim, bool save=False, str save_name="void", bool show=False)
Make a plot showing Omega for BWD.
- float [auxiliary.tau_syst](#) (float f_0, float f_1, float K, float s_in_Myr)
Calculates tau, the time it takes a binary with K to evolve from f_0 to f_1 (GW frequencies).
- float [auxiliary.determine_upper_freq](#) (float nu_low, float evolve_time, float K, bool DEBUG=False)
Determines upper ORBITAL frequency for a binary with K, starting from nu_0, evolving over evolve_time.

Variables

- tuple [auxiliary.s_in_Myr](#) = (u.Myr).to(u.s)

6.5.1 Detailed Description

This module contains auxiliary functions that are used in the main code.

Author

Seppe Staelens

Date

2024-07-24

6.6 /home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/RedshiftInterpolator.py File Reference

Classes

- class [RedshiftInterpolator.RedshiftInterpolator](#)
This class is used to quickly determine the redshift at a given age of the Universe.

Namespaces

- namespace [RedshiftInterpolator](#)

This module contains the class [RedshiftInterpolator](#).

6.7 /home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_Dwarf_AGWB/src/modules/SFH.py File Reference ↩

This file contains the functions to determine the star formation rate.

Namespaces

- namespace [SFH](#)

Functions

- [SFH.representative_SFH](#) (float age, [ri.RedshiftInterpolator](#) redshift_interpolator, float Delta_t=0., int SFH_num=1, float max_z=8.)
Determines an appropriate value for the star formation rate at a given age.
- float [SFH.SFH_MD](#) (float z)
Star formation history from [Madau, Dickinson 2014].
- float [SFH.SFH2](#) (float z)
Made up star formation history.
- float [SFH.SFH3](#) (float z)
Made up star formation history.
- float [SFH.SFH4](#) (float z)
Made up star formation history.

6.7.1 Detailed Description

This file contains the functions to determine the star formation rate.

Date

2024-07-24

The file contains the functions to determine the star formation rate. The function `representative_SFH` determines an appropriate value for the star formation rate at a given age. It allows for an optional additional time delay, due to a delay in formation of the binary, or if time is required to move to the correct frequency bin. The functions `SFH_MD`, `SFH2`, `SFH3`, and `SFH4` are star formation histories that can be selected.

Author

Seppe Staelens

6.8 /home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_↵ Dwarf_AGWB/src/modules/SimModel.py File Reference

Classes

- class [SimModel.SimModel](#)

! This class contains information about the run that needs to be shared over the different subroutines.

Namespaces

- namespace [SimModel](#)

This module contains the class [SimModel](#).

6.9 /home/seppe/Documents/data/Papers/AnA.683.A139(2024)/White_↵ Dwarf_AGWB/src/num_syst.py File Reference

Classes

- class [num_syst.sim_model](#)

MODEL CLASS #####.

Namespaces

- namespace [num_syst](#)

Functions

- [num_syst.get_width_z_shell_from_z](#) (z_vals)
AUXILIARY FUNCTIONS #####.
- [num_syst.SFH](#) (z)
Star formation history from [Madau, Dickinson 2014].
- [num_syst.tau_syst](#) (f_0, f_1, K)
Calculates tau, the time it takes a binary with K to evolve from f_0 to f_1 (GW frequencies).
- [num_syst.representative_SFH](#) (age, Delta_t, SFH_num, max_z)
Looks for a representative value of the [SFH](#) given the age of the system, and an additional time delay in reaching the bin.
- [num_syst.get_z_fast](#) (age)
- [num_syst.determine_upper_freq](#) (nu_low, evolve_time, K)
Determines upper ORBITAL frequency for a binary with K, starting from nu_0, evolving over evolve_time.
- [num_syst.num_merge_bins](#) (model1, model2, data, tag)
- [num_syst.main](#) ()
ACTUAL MAIN FUNCTION #####.

Variables

- [num_syst.size](#)
- [num_syst.titlesize](#)
- [num_syst.labelsize](#)
- [num_syst.fontsize](#)
- tuple [num_syst.s_in_Myr](#) = (u.Myr).to(u.s)
- [num_syst.z_at_val_data](#) = pd.read_csv("../Data/z_at_age.txt", names=["age", "z"], header=1)
LOAD Z_AT_VALUE FILE #####.
- [num_syst.interp_age](#)
- [num_syst.interp_z](#)

Index

/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/WhiteDwarf_AGWB/src/GWB.py, 35
calculate_cosmology_from_z
/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/WhiteDwarf_AGWB/src/modules/RedshiftInterpolator.py, 38
calculate_f_bins
/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/WhiteDwarf_AGWB/src/modules/SFH.py, 39
SimModel.SimModel, 29
/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/WhiteDwarf_AGWB/src/modules/SimModel.py, 40
SimModel.SimModel, 30
/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/WhiteDwarf_AGWB/src/modules/add_birth.py, 36
num_syst.sim_model, 26
/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/WhiteDwarf_AGWB/src/modules/add_bulk.py, 36
category
/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/WhiteDwarf_AGWB/src/modules/add_merge.py, 37
category
/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/WhiteDwarf_AGWB/src/modules/auxiliary.py, 37
SimModel.SimModel, 30
/home/seppe/Documents/data/Papers/AnA.683.A139(2024)/WhiteDwarf_AGWB/src/num_syst.py, 40
determine_upper_freq
auxiliary, 11
num_syst, 16
__init__
num_syst.sim_model, 25
RedshiftInterpolator.RedshiftInterpolator, 23
SimModel.SimModel, 29
action
GWB, 14
add_birth, 7
add_birth, 7
add_bulk, 8
add_bulk, 8
add_merge, 8
add_merge, 8
ages
num_syst.sim_model, 26
SimModel.SimModel, 30
auxiliary, 9
calc_parabola_vertex, 10
determine_upper_freq, 11
get_bin_factors, 11
get_width_z_shell_from_z, 11
make_Omega_plot_unnorm, 12
Omega, 12
parabola, 13
s_in_Myr, 14
tau_syst, 13
calc_parabola_vertex
auxiliary, 10
calculate_cosmology
num_syst.sim_model, 25
calculate_cosmology_from_T
SimModel.SimModel, 29
calculate_cosmology_from_z
calculate_f_bins
WhiteDwarf_AGWB/src/modules/SFH.py, 39
WhiteDwarf_AGWB/src/modules/SimModel.py, 40
WhiteDwarf_AGWB/src/modules/add_birth.py, 36
WhiteDwarf_AGWB/src/modules/add_bulk.py, 36
WhiteDwarf_AGWB/src/modules/add_merge.py, 37
WhiteDwarf_AGWB/src/modules/auxiliary.py, 37
WhiteDwarf_AGWB/src/num_syst.py, 40
determine_upper_freq
auxiliary, 11
num_syst, 16
dT
SimModel.SimModel, 31
f_bin_factors
SimModel.SimModel, 31
f_bins
num_syst.sim_model, 26
SimModel.SimModel, 31
f_plot
num_syst.sim_model, 26
SimModel.SimModel, 31
fontsize
GWB, 14
num_syst, 17
get_bin_factors
auxiliary, 11
get_width_z_shell_from_z
auxiliary, 11
num_syst, 16
get_z_fast
num_syst, 16
RedshiftInterpolator.RedshiftInterpolator, 24
GWB, 14
action, 14
category, 14
fontsize, 14
labels, 15
main, 14
s_in_Myr, 15
size, 15

- titlesize, 15
- INTEG_MODE
 - SimModel.SimModel, 31
- interp_age
 - num_syst, 17
 - RedshiftInterpolator.RedshiftInterpolator, 24
- interp_z
 - num_syst, 17
 - RedshiftInterpolator.RedshiftInterpolator, 24
- labelsiz
 - GWB, 15
 - num_syst, 18
- light_speed
 - SimModel.SimModel, 31
- log_f_high
 - num_syst.sim_model, 26
 - SimModel.SimModel, 31
- log_f_low
 - num_syst.sim_model, 26
 - SimModel.SimModel, 31
- main
 - GWB, 14
 - num_syst, 16
- make_Omega_plot_unnorm
 - auxiliary, 12
- max_z
 - num_syst.sim_model, 26
 - SimModel.SimModel, 31
- N
 - num_syst.sim_model, 26
- N_freq
 - SimModel.SimModel, 32
- N_int
 - SimModel.SimModel, 32
- N_z
 - num_syst.sim_model, 26
- num_merge_bins
 - num_syst, 16
- num_syst, 15
 - determine_upper_freq, 16
 - fontsize, 17
 - get_width_z_shell_from_z, 16
 - get_z_fast, 16
 - interp_age, 17
 - interp_z, 17
 - labelsiz, 18
 - main, 16
 - num_merge_bins, 16
 - representative_SF
H, 17
 - s_in_Myr, 18
 - SFH, 17
 - size, 18
 - tau_syst, 17
 - titlesiz, 18
 - z_at_val_data, 18
- num_syst.sim_model, 24
 - __init__, 25
 - ages, 26
 - calculate_cosmology, 25
 - calculate_f_bins, 25
 - calculate_z_bins, 26
 - f_bins, 26
 - f_plot, 26
 - log_f_high, 26
 - log_f_low, 26
 - max_z, 26
 - N, 26
 - N_z, 26
 - SFH_num, 27
 - z_bins, 27
 - z_list, 27
 - z_time_since_max_z, 27
 - z_widths, 27
- Omega
 - auxiliary, 12
- parabola
 - auxiliary, 13
- RedshiftInterpolator, 18
 - RedshiftInterpolator.RedshiftInterpolator, 23
 - __init__, 23
 - get_z_fast, 24
 - interp_age, 24
 - interp_z, 24
- representative_SF
H
 - num_syst, 17
 - SFH, 19
- s_in_Myr
 - auxiliary, 14
 - GWB, 15
 - num_syst, 18
- SAVE_FIG
 - SimModel.SimModel, 32
- set_mode
 - SimModel.SimModel, 30
- SFH, 19
 - num_syst, 17
 - representative_SF
H, 19
 - SFH2, 19
 - SFH3, 21
 - SFH4, 21
 - SFH_MD, 21
- SFH2
 - SFH, 19
- SFH3
 - SFH, 21
- SFH4
 - SFH, 21
- SFH_MD
 - SFH, 21
- SFH_num

- num_syst.sim_model, [27](#)
 - SimModel.SimModel, [32](#)
- SimModel, [22](#)
- SimModel.SimModel, [27](#)
 - __init__, [29](#)
 - ages, [30](#)
 - calculate_cosmology_from_T, [29](#)
 - calculate_cosmology_from_z, [29](#)
 - calculate_f_bins, [29](#)
 - calculate_T_bins, [30](#)
 - calculate_z_bins, [30](#)
 - DEBUG, [30](#)
 - dT, [31](#)
 - f_bin_factors, [31](#)
 - f_bins, [31](#)
 - f_plot, [31](#)
 - INTEG_MODE, [31](#)
 - light_speed, [31](#)
 - log_f_high, [31](#)
 - log_f_low, [31](#)
 - max_z, [31](#)
 - N_freq, [32](#)
 - N_int, [32](#)
 - SAVE_FIG, [32](#)
 - set_mode, [30](#)
 - SFH_num, [32](#)
 - T0, [32](#)
 - T_bins, [32](#)
 - T_list, [32](#)
 - T_range, [32](#)
 - TEST_FOR_ONE, [32](#)
 - z_bins, [32](#)
 - z_list, [33](#)
 - z_time_since_max_z, [33](#)
 - z_widths, [33](#)
- size
 - GWB, [15](#)
 - num_syst, [18](#)
- T0
 - SimModel.SimModel, [32](#)
- T_bins
 - SimModel.SimModel, [32](#)
- T_list
 - SimModel.SimModel, [32](#)
- T_range
 - SimModel.SimModel, [32](#)
- tau_syst
 - auxiliary, [13](#)
 - num_syst, [17](#)
- TEST_FOR_ONE
 - SimModel.SimModel, [32](#)
- titlesize
 - GWB, [15](#)
 - num_syst, [18](#)
- z_at_val_data
 - num_syst, [18](#)
- z_bins
 - num_syst.sim_model, [27](#)
 - SimModel.SimModel, [32](#)
- z_list
 - num_syst.sim_model, [27](#)
 - SimModel.SimModel, [33](#)
- z_time_since_max_z
 - num_syst.sim_model, [27](#)
 - SimModel.SimModel, [33](#)
- z_widths
 - num_syst.sim_model, [27](#)
 - SimModel.SimModel, [33](#)