

---

# **Fruitbat Documentation**

***Release 2.0.1***

**Adam Batten**

**Jul 06, 2021**



<b>1</b>	<b>Getting started</b>	<b>3</b>
1.1	Installation . . . . .	3
1.2	Simple Example . . . . .	4
<b>2</b>	<b>Using Fruitbat</b>	<b>5</b>
2.1	Frb Class . . . . .	5
2.2	Custom Lookup Tables . . . . .	8
<b>3</b>	<b>Methods and Cosmology</b>	<b>11</b>
3.1	Methods . . . . .	11
3.2	Cosmology . . . . .	13
<b>4</b>	<b>FAQ</b>	<b>15</b>
4.1	Why is it called fruitbat? . . . . .	15
4.2	I'd rather use the NE2001 model. Is this compatible with Fruitbat? . . . . .	15
<b>5</b>	<b>NE2001 Installation Instructions</b>	<b>17</b>
5.1	Installing the NE2001 model . . . . .	17
5.2	Using the NE2001 model . . . . .	17
<b>6</b>	<b>Guidelines</b>	<b>19</b>
6.1	Reference . . . . .	19
6.2	Issues and Contributing . . . . .	19
6.3	License . . . . .	20
<b>7</b>	<b>fruitbat.Frb</b>	<b>21</b>
<b>8</b>	<b>fruitbat.methods</b>	<b>31</b>
<b>9</b>	<b>fruitbat.cosmologies</b>	<b>33</b>
<b>10</b>	<b>fruitbat.catalogue</b>	<b>37</b>
<b>11</b>	<b>fruitbat.table</b>	<b>39</b>
11.1	table . . . . .	39
<b>12</b>	<b>fruitbat.plot</b>	<b>41</b>

<b>13 fruitbat.utils</b>	<b>43</b>
<b>Python Module Index</b>	<b>47</b>
<b>Index</b>	<b>49</b>

*FRUITBAT* is an open source Python package designed to assist the estimation of redshifts, energies and the galactic dispersion measure contributions of fast radio bursts.

*FRUITBAT* generates and utilises ‘look-up’ tables of existing dispersion measure-redshift relations found in the literature (Ioka 2003, Inoue 2004, Zhang 2018) in conjunction with parameters from both the WMAP and Planck missions. *FRUITBAT* also utilises the YMW16 galactic dispersion measure model to estimate the dispersion measure contribution due to the Milky Way. However it is also possible to use the NE2001 model if the python port has been installed (See the NE2001 installation instructions).

As a user you can independantly choose the dispersion measure-reshift relation and the cosmological parameters, or define your own relation, create new cosmologies and generate custom look-up tables for *FRUITBAT*.

*FRUITBAT* is installable via `pip` (see [Getting Started](#)) or the source code is made available [here](#).

If you use **Fruitbat** in your research, please add the acknowledgement statement “Some of the results of this paper have been derived using the *FRUITBAT* package” and cite the JOSS paper.

```
@ARTICLE{2019JOSS....4.1399B,
  author = {{Batten}, Adam},
  title = "{Fruitbat: A Python Package for Estimating Redshifts of Fast Radio_
↪Bursts}",
  journal = {The Journal of Open Source Software},
  keywords = {Astrophysics - Instrumentation and Methods for Astrophysics, ↪
↪Astrophysics - High Energy Astrophysical Phenomena},
  year = "2019",
  month = "May",
  volume = {4},
  number = {37},
  pages = {1399},
  doi = {10.21105/joss.01399},
archivePrefix = {arXiv},
  eprint = {1905.04294},
  primaryClass = {astro-ph.IM},
  adsurl = {https://ui.adsabs.harvard.edu/abs/2019JOSS....4.1399B},
  adsnote = {Provided by the SAO/NASA Astrophysics Data System}
}
```



### 1.1 Installation

You can install the latest release of **fruitbat** from [PyPi](#) by running the following:

```
pip install fruitbat
```

You can install the latest development version of **fruitbat** by cloning the [repository](#):

```
git clone https://github.com/abatten/fruitbat
cd fruitbat
pip install .
```

If you are installing the latest development version of *FRUITBAT* then you will also need to install git-lfs. Instructions for installing git-lfs for your operating system can be found [here](#).

#### 1.1.1 Requirements

Below are the listed requirements for running **fruitbat** and the purpose for each requirement.

- **numpy**: Array manipulation
- **scipy**: Modules for integration and interpolation
- **astropy**: Modules for cosmology, coordinates, constants and units
- **matplotlib**: Modules for plotting
- **pandas**: Reading csv files from FRBCAT
- **pyymw16**: Python wrapper for YMW16 galactic dispersion measure model.
- **e13tools**: Utility tools for writing docstrings.

## 1.1.2 Running Tests

**Fruitbat** uses `pytest` to perform all its testing. If you would like to run the tests yourself you will need to install all the packages in `requirements_dev`. The easiest way of doing this is by cloning the [source directory](#), install all requirements and running `pytest`:

```
git clone https://github.com/abatten/fruitbat
cd fruitbat
pip install -r requirements_dev.txt
pytest
```

Pytest should take around 5 minutes.

## 1.2 Simple Example

A detailed explanation of this example can be viewed at [Using Fruitbat](#).

```
>>> import fruitbat

# Create a Frb Object with DM and Galactic Coordinates
>>> FRB180110 = fruitbat.Frb(715.7, gl="7.8", gb="-51.9", name="FRB180110")

# Calculate the DM contribution from the Milky Way
>>> FRB180110.calc_dm_galaxy()

# Calculate the Redshift of the FRB using the relation from Zhang (2018)
>>> FRB180110.calc_redshift(method="Zhang2018", cosmology="Planck18")
```



Here is the guide of how to use the **fruitbat** package.

### 2.1 Frb Class

Most calculations in **fruitbat** are centred around the class *Frb*. Hence you will need to define a *fruitbat.Frb* object for each FRB. The minimum required to define a *Frb* is the observed dispersion measure (DM).

```
>>> import fruitbat
>>> frb = fruitbat.Frb(635.1)
```

It's also possible to 'name' the FRB using the keyword *name*, which can be useful when calculating redshifts of many FRBs at once.

```
>>> import fruitbat
>>> frb = fruitbat.Frb(635.1, name="FRB190229")
```

To see a full list of possible parameters visit the API documentation for *Frb*.

#### 2.1.1 Redshift Estimation

To estimate the redshift of the FRB, use the method *calc\_redshift()*. Unless otherwise specified this assumes that the entire DM contribution is due to the IGM. To see how to account for the Milky Way and host galaxy contributions to the DM see the sections on *Galactic Dispersion Measure* and *Host Galaxy Dispersion Measure* respectively.

```
>>> frb = fruitbat.Frb(635.1)
>>> frb.calc_redshift()
<Quantity 0.63199287>
```

The *calc\_redshift()* also has keywords to select alternative IGM models and cosmologies when estimating the redshift of the FRB (Default method and cosmology are *Inoue2004* and *Planck2018* respectively).

```
>>> frb.calc_redshift(method="Zhang2018")
<Quantity 0.70986024>

>>> frb.calc_redshift(method="Ioka2003", cosmology="Planck13")
<Quantity 0.52776778>
```

Currently available methods in **fruitbat** include: Ioka2003, Inoue2004, Zhang2018. Currently available cosmologies include: WMAP5, WMAP7, WMAP9, Planck13, Planck15, Planck18, EAGLE. It should be of note that EAGLE uses the Planck13 cosmology but is listed here for convenience.

## Galactic Dispersion Measure

When estimating the redshift of an FRB, it is necessary to account for the Milky Way's contribution to the observed DM. Depending on the location on the sky, this contribution can be as low as  $30 \text{ pc cm}^{-3}$  out of the disk of the Milky Way and exceeding  $1000 \text{ pc cm}^{-3}$  through the disk. Without accurately accounting for this contribution, redshift estimates of FRB would be significantly higher than their physical redshift.

Within **fruitbat** there are three main ways to account for the galactic contribution: `calc_dm_galaxy()`, `dm_galaxy` or `dm_excess`.

### Method 1: calc\_dm\_galaxy()

The first and easiest way to account for the galactic contribution is to provide the sky coordinates of the FRB when instantiating the object, then call `calc_dm_galaxy()`. The `calc_dm_galaxy()` method of `Frb` estimates the total DM contribution due to the Milky Way along the line of sight of the FRB using the YMW16 galactic free electron model.

```
>>> frb = fruitbat.Frb(635.1, gl="35.1", gb="12.5")
>>> frb.calc_dm_galaxy()
<Quantity 114.27922821 pc / cm3>

>>> frb = fruitbat.Frb(635.1, raj="18:10:34.8668", decj="7:33:35.9289")
>>> frb.calc_dm_galaxy()
<Quantity 114.27922821 pc / cm3>
```

The sky coordinates can be in either ICRS or Galactic units. The `calc_dm_galaxy()` method will calculate the `dm_excess` by subtracting the estimated `dm_galaxy` from the observed DM. After calculating `dm_galaxy`, calling `calc_redshift()` will automatically use the calculated `dm_excess` to estimate the redshift.

```
>>> frb.calc_redshift()
<Quantity 0.52244866>
```

### Method 2: dm\_galaxy

The second method to account for the galactic dispersion measure is to provide a value of `dm_galaxy`. This value could be calculated from another galactic dispersion measure model such as the NE2001 model, which has not been implemented in **fruitbat** yet.

```
>>> frb = fruitbat.Frb(635.1, dm_galaxy=114.28)
>>> frb.calc_redshift()
<Quantity 0.52244791>
```

### Method 3: `dm_excess`

The third and final method is to directly specify the `dm_excess`. This provides the value that will be used for the redshift calculation directly.

```
>>> frb = fruitbat.Frb(635.1, dm_excess=520.82)
>>> frb.calc_redshift()
<Quantity 0.52244791>
```

### Host Galaxy Dispersion Measure

It's common to assume that the host galaxy contributes nothing to the observed dispersion measure, however this is unlikely to be true. Use the parameter `dm_host_est` to provide an estimate of the contribution to the observed dispersion measure due to the FRB host and set specify 'subtract\_host=True' in the method `calc_redshift()`.

```
>>> frb = fruitbat.Frb(635.1, gl="35.1", gb="12.5", dm_host_est=64.1)
>>> frb.calc_dm_galaxy()
>>> frb.calc_redshift()
<Quantity 0.52244866>
>>> frb.calc_redshift(subtract_host=True)
<Quantity 0.46077303>
```

## 2.1.2 Calculating Distances

Other than redshift, **fruitbat** has two other distance functions however both require the redshift to be calculated first. These distance functions are `calc_comoving_distance()` and `calc_luminosity_distance()`

### Comoving Distance

To calculate the comoving distance use `calc_comoving_distance()` after calculating the redshift.

$$d_C(z) = \frac{c}{H_0} \int_0^z \frac{dz'}{\sqrt{\Omega_r(1+z')^4 + \Omega_m(1+z')^3 + \Omega_\Lambda}}$$

```
>>> frb = fruitbat.Frb(635.1, gl="35.1", gb="12.5")
>>> frb.calc_dm_galaxy()
>>> frb.calc_redshift()
>>> frb.calc_comoving_distance()
<Quantity 2020.29768846 Mpc>
```

### Luminosity Distance

To calculate the luminosity distance use `calc_luminosity_distance()` after calculating the redshift.

$$d_L(z) = (1+z)d_C(z)$$

```
>>> frb = fruitbat.Frb(635.1, gl="35.1", gb="12.5")
>>> frb.calc_dm_galaxy()
>>> frb.calc_redshift()
>>> frb.calc_luminosity_distance()
<Quantity 3075.79950018 Mpc>
```

### 2.1.3 Calculating Energy

The method `calc_energy()` estimates the upper limit to the isotropic energy of the FRB. To use the method `calc_energy()`, the FRB requires the following properties: `fluence` and `obs_freq_central`. `fluence` can be calculated by providing `peak_flux` and `width`.

```
>>> frb.fluence = 2.0
>>> frb.obs_freq_central = 1600
>>> frb.calc_energy()
<Quantity 2.37921847e+40 erg>
```

### 2.1.4 Calculating Luminosity

The method `calc_luminosity()` estimates the upper limit to the isotropic peak luminosity of the FRB. To use the method `calc_luminosity()`, the FRB requires the following properties: `peak_flux` and `obs_freq_central`.

```
>>> frb.peak_flux = 0.1
>>> frb.obs_freq_central = 1600
>>> frb.calc_luminosity()
<Quantity 1.81111898e+42 erg / s>
```

## 2.2 Custom Lookup Tables

### 2.2.1 Custom Methods

```
>>> def simple_dm(z):
    dm = 1200 * z
    return dm
>>> fruitbat.add_method("simple_dm", simple_dm)
>>> fruitbat.available_methods()
['Ioka2003', 'Inoue2004', 'Zhang2018', 'simple_dm']
```

### 2.2.2 Custom Cosmologies

```
>>> params = {"H0": 72.4, "Om0": 0.26}
>>> new_cosmology = fruitbat.cosmologies.create_cosmology(parameters=params)
>>> fruitbat.add_cosmology("new_cosmology", new_cosmology)
>>> fruitbat.available_cosmologies()
['WMAP5', 'WMAP7', 'WMAP9', 'Planck13', 'Planck15', 'Planck18', 'EAGLE', 'new_
↪cosmology']
```

### 2.2.3 Custom Table

To create a custom lookup table you first need to define a custom method and add that method to the list of available methods using `add_method()`. Then you can use `create()` to generate a lookup table of that method.

```
>>> def simple_dm(z):  
    dm = 1200 * z  
    return dm  
>>> fruitbat.add_method("simple_dm", simple_dm)  
>>> fruitbat.table.create("simple_dm")  
>>> frb = fruitbat.Frb(1200)  
>>> frb.calc_redshift(method="simple_dm")  
<Quantity 1.>
```



---

## Methods and Cosmology

---

When using **fruitbat** there are different methods and cosmologies built-in from which you can choose. Below is a brief description of how each method is calculated and the parameters for each cosmology.

### 3.1 Methods

Currently there are three built-in dispersion measure-redshift relations. In **fruitbat** these are referred to as ‘methods’. To calculate the look-up tables **fruitbat** uses the equations described below for each method to calculate the dispersion measure (DM) for a given redshift ( $z$ ).

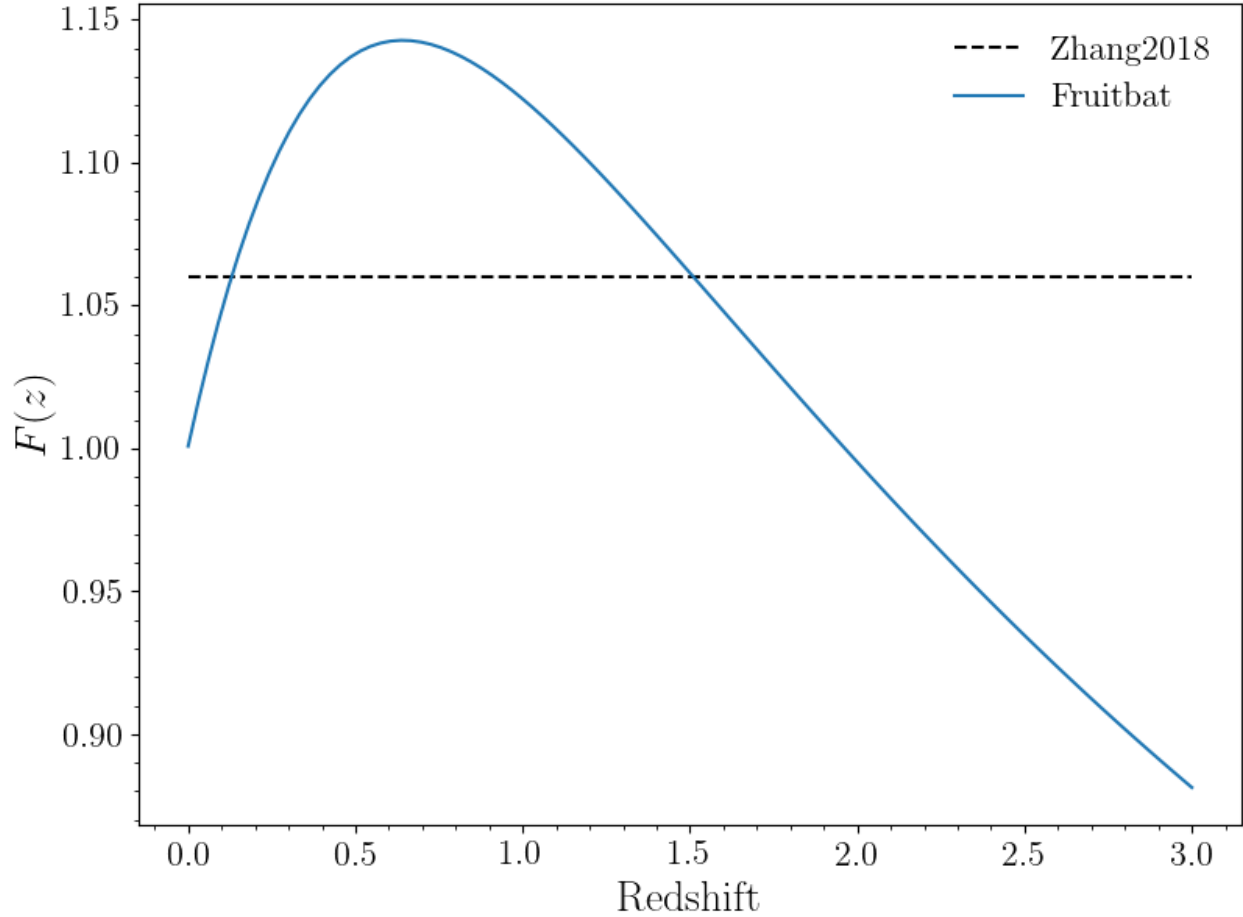
#### 3.1.1 $F(z)$ Integral

The methods 'Ioka2003', 'Inoue2004', 'Zhang2018' all have a common integral  $F(z)$ .

$$F(z) = \int_0^z \frac{1+z}{[\Omega_m(1+z)^3 + \Omega_\Lambda(1+z)^{3(1-w)}]^{1/2}} dz$$

Where  $z$  is the redshift,  $\Omega_m$  is the cosmic matter density,  $\Omega_\Lambda$  is cosmic dark energy density and  $w$  characterises the dark energy equation of state. This is typically assumed to have a constant value of  $\sim 1.06$  which introduces an error of approximately 6% for  $z < 2$ .

The lookup tables in **fruitbat** explicitly solve this integral for each redshift to when calculating a dispersion measure. See the Figure below for a comparison between the assumed value and the integral in **fruitbat**.



### 3.1.2 Ioka 2003

The [Ioka \(2003\)](#) method assumes that all baryons in the Universe are fully ionised and that there is a 1-to-1 relation between electrons and baryons. i.e. The number of free electrons in the Universe is the same as the number of baryons. The DM at a given redshift using the Ioka method is calculated as follows:

$$\begin{aligned} \text{DM} &= \frac{3cH_0\Omega_b}{8\pi Gm_p} F(z) \\ &= 1108 \text{pc cm}^{-3} F(z) \end{aligned}$$

Where  $c$  is the speed of light,  $H_0$  is the Hubble constant,  $\Omega_b$  is the cosmic baryon density,  $G$  is the gravitational constant,  $m_p$  is the mass of the proton and  $F(z)$  is the integral defined earlier.

### 3.1.3 Inoue 2004

The [Inoue \(2004\)](#) method assumes that hydrogen is fully ionised and helium is singly ionised. The DM at a given redshift using the Inoue method is calculated as follows.

$$\begin{aligned} \text{DM} &= 9.2 \times 10^{-10} cH_0\Omega_b F(z) \\ &= 908 \text{pc cm}^{-3} F(z) \end{aligned}$$

Where  $c$  is the speed of light,  $H_0$  is the Hubble constant,  $\Omega_b$  is the cosmic baryon density and  $F(z)$  is the integral defined earlier. The factor of  $9.2 \times 10^{-10}$  comes from estimating the number of free electrons at high redshifts from models of reionisation.



### 3.1.4 Zhang 2018

The [Zhang \(2018\)](#) method assumes that all baryons in the Universe are fully ionised and that there is a 0.875-to-1 ratio between electrons and baryons, and that 85% of baryons are in the intergalactic medium. The DM at a given redshift using the Zhang method is calculated as follows:

$$\begin{aligned} \text{DM} &= \frac{3cH_0\Omega_b\chi f_{igm}}{8\pi Gm_p} F(z) \\ &= 805 \text{pc cm}^{-3} \left( \frac{f_{igm}}{0.83} \right) \left( \frac{\chi}{0.875} \right) F(z) \end{aligned}$$

Where  $c$  is the speed of light,  $H_0$  is the Hubble constant,  $\Omega_b$  is the cosmic baryon density,  $G$  is the gravitational constant,  $m_p$  is the mass of the proton,  $\chi$  is the free electron per baryon in the Universe,  $f_{igm}$  is the fraction of baryons in the intergalactic medium and  $F(z)$  is the integral defined earlier.

$\chi$  is calculated as follows:

$$\chi(z) = \frac{3}{4}y_1\chi_{e,H}(z) + \frac{1}{8}y_2\chi_{e,He}(z)$$

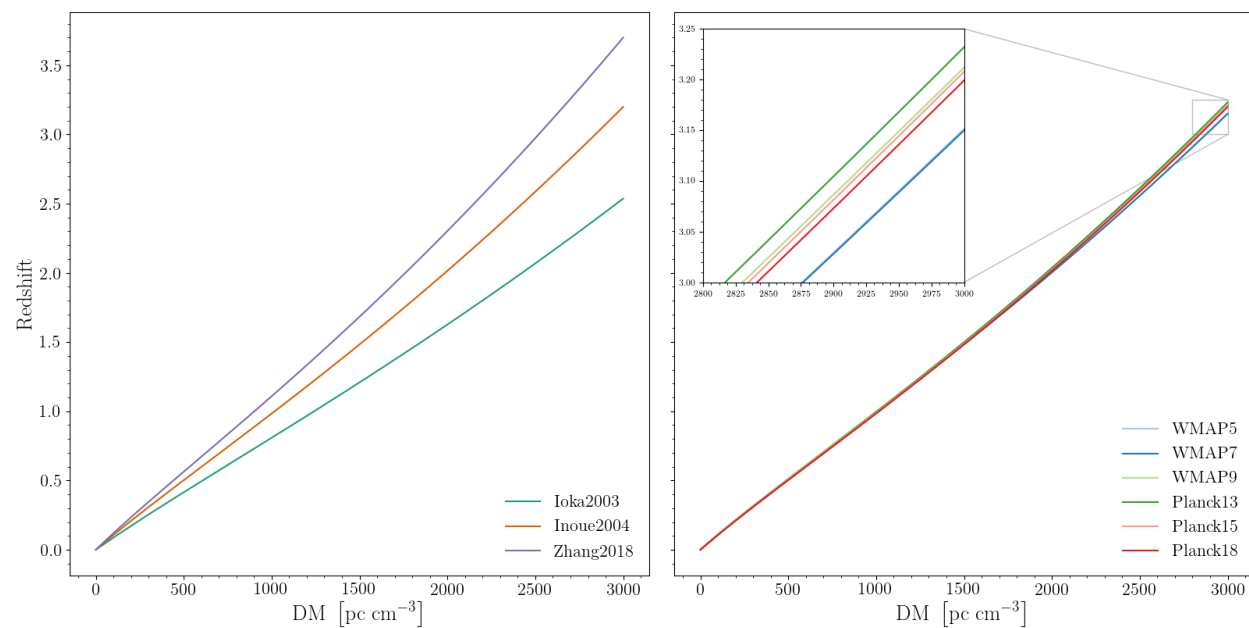
Where  $\chi_{e,H}$  and  $\chi_{e,He}$  denote the ionisation fraction of hydrogen and helium respectively and  $y_1 \sim y_2 \sim 1$  denote the possible slight deviation from the 3/4 - 1/4 split of hydrogen and helium abundance in the Universe. Assuming that hydrogen and helium are both ionised, then  $\chi(z) \sim 0.875$ .

## 3.2 Cosmology

Each method in **fruitbat** has a list of pre-calculated lookup tables with different cosmologies. The table below lists the parameters that are used for each cosmology.

Cosmological Parameters					
Keyword	$H_0$	$\Omega_b$	$\Omega_m$	$\Omega_\Lambda$	$w$
'WMAP5'	70.2	0.0459	0.277	0.723	-1
'WMAP7'	70.4	0.0455	0.272	0.728	-1
'WMAP9'	69.32	0.04628	0.2865	0.7134	-1
'Planck13'	67.77	0.0483	0.3071	0.6914	-1
'Planck15'	67.74	0.0486	0.3075	0.6910	-1
'Planck18'	67.66	0.04897	0.3111	0.6874	-1

Below is a figure comparing the different methods and cosmologies in **fruitbat**. The left figure shows how the different methods compare assuming a 'Planck18' cosmology.. The right figure shows how the 'Inoue2004' method changes with different assumed cosmologies.



## 4.1 Why is it called fruitbat?

A lot of time was spent trying to figure out a ‘nice’ acronym, however none of the potential names seemed to fit. So I decided to just name it after an animal. After searching through lists of Australian animals I remembered the family of fruitbats (grey-headed flying foxes) that live in the tree in the front of my house. Fruitbat had the letters FRB in order and their orange necks reminded me of waterfall plots so I decided to go with the name Fruitbat.

## 4.2 I’d rather use the NE2001 model. Is this compatible with Fruitbat?

Yes! However you will need to install the python port of the NE2001 model. You can find installation and usage instructions [here](#).

Alternatively, you can calculate the galactic DM using any version of the NE2001 model first and then use that value for *dm\_galaxy*.

```
>>> ne2001_dm = 34.5
>>> import fruitbat
>>> FRB = fruitbat.Frb(dm=325, dm_galaxy=ne2001_dm, gl="32.4", gb="16.2")
```



---

## NE2001 Installation Instructions

---

When you install *FRUITBAT* it automatically installs the YMW16 galactic electron density model `pyywm16` from PyPi. However, if you have the python port of the NE2001 model by Ben Baror and JXP you can use it with *FRUITBAT* to estimate the galactic contribution to dispersion measure.

The python port that is compatible with *FRUITBAT* is found at: <https://github.com/FRBs/ne2001>.

You will need to download and install the NE2001 model to use it.

### 5.1 Installing the NE2001 model

```
>>> git clone https://github.com/FRBs/ne2001
>>> cd ne2001
>>> pip install .
```

### 5.2 Using the NE2001 model

After installing the NE2001 model, it behaves exactly the same as the YMW16 model. To specify that you want to use the NE2001 model pass “*ne2001*” as a keyword in `calc_dm_galaxy()`.

```
>>> import fruitbat
>>> FRB190523 = fruitbat.Frb(760.8, gl="117.03", gb="44")
>>> dm_galaxy_ymw16 = FRB190523.calc_dm_galaxy(model="ymw16")
>>> dm_galaxy_ne2001 = FRB190523.calc_dm_galaxy(model="ne2001")
>>> print(dm_galaxy_ymw16, dm_galaxy_ne2001)
29.88017464 pc / cm3    36.87013932 pc / cm3
```



## 6.1 Reference

If you use **Fruitbat** in your research, please add the acknowledgement statement “Some of the results of this paper have been derived using the *FRUITBAT* package” and cite the JOSS paper.

```
@ARTICLE{2019JOSS....4.1399B,
  author = {{Batten}, Adam},
  title = "{Fruitbat: A Python Package for Estimating Redshifts of Fast Radio_
↪Bursts}",
  journal = {The Journal of Open Source Software},
  keywords = {Astrophysics - Instrumentation and Methods for Astrophysics, ↪
↪Astrophysics - High Energy Astrophysical Phenomena},
  year = "2019",
  month = "May",
  volume = {4},
  number = {37},
  pages = {1399},
  doi = {10.21105/joss.01399},
archivePrefix = {arXiv},
  eprint = {1905.04294},
  primaryClass = {astro-ph.IM},
  adsurl = {https://ui.adsabs.harvard.edu/abs/2019JOSS....4.1399B},
  adsnote = {Provided by the SAO/NASA Astrophysics Data System}
}
```

## 6.2 Issues and Contributing

If there is a feature of *FRUITBAT* that currently does not exist, but you would like it to, you can contribute by opening a [Github Issue](#) and outlining the feature. Similar to contributing, if you find a problem with *FRUITBAT* or are having difficulties using *FRUITBAT* please do not hesitate to open a [Github Issue](#).

## 6.3 License

BSD 3-Clause License

Copyright (c) 2019, Adam Batten  
All rights reserved.

Redistribution and use in source and binary forms, with or without  
modification, are permitted provided that the following conditions are met:

- \* Redistributions of source code must retain the above copyright notice, this  
list of conditions and the following disclaimer.
- \* Redistributions in binary form must reproduce the above copyright notice,  
this list of conditions and the following disclaimer in the documentation  
and/or other materials provided with the distribution.
- \* Neither the name of the copyright holder nor the names of its  
contributors may be used to endorse or promote products derived from  
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"  
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE  
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE  
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE  
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL  
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR  
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER  
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,  
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE  
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.



```
class fruitbat.Frb(dm, name=None, raj=None, decj=None, gl=None, gb=None, dm_galaxy=0.0,
                    dm_excess=None, z_host=None, dm_host_est=0.0, dm_host_loc=0.0,
                    dm_index=None, scatt_index=None, snr=None, width=None, peak_flux=None,
                    fluence=None, obs_bandwidth=None, obs_freq_central=None, utc=None,
                    **kwargs)
```

Create a *Frb* object using the observed properties of a FRB including dispersion measure (DM) and its sky coordinates. This class utilises various DM-redshift relations as well as the YMW16 galactic DM model in the analysis.

A FRB can be defined simply with a single *dm* value. This should be the observed DM prior to subtracting the Milky Way, however if you do not provide any additional information, it is assumed that this value is also the *dm\_excess* (DM - Galactic DM). You can get the estimated redshift by using the method *calc\_redshift()*.

To get a more accurate distance estimate you can account for the contribution due to the Milky Way by supplying *dm\_galaxy* or by giving the coordinates of the FRB in ICRS (*raj*, *decj*) or Galactic (*gl*, *gb*) coordinates and calling the method *calc\_dm\_galaxy()* before calling *calc\_redshift()*.

**Parameters** *dm* (*float*) – The observed dispersion measure of the FRB. This is without Milky Way or host galaxy subtraction. Units: pc cm<sup>-3</sup>

### Keyword Arguments

- **name** (*str or None, optional*) – The name of the frb object. Default: *None*
- **raj** (*str or None, optional*) – The right ascension in J2000 coordinates of the best estimate of the FRB position. By default should be given in the ICRS frame. An alternative frame can be specified by also including the keyword *frame*. e.g. *frame='fk5'*. Default: *None*
- **decj** (*str or None, optional*) – The declination in J2000 coordinates of the best estimate of the FRB position. By default should be given in the ICRS frame. An alternative frame can be specified by also including the keyword *frame*. e.g. *frame='fk5'*. Default: *None*
- **gl** (*str or None, optional*) – The Galactic longitude in degrees of the best estimate of the FRB position. Default: *None*

- **gb** (*str or None, optional*) – The Galactic latitude in degrees of the best estimate of the FRB position. Default: *None*

#### Other Parameters

- **dm\_galaxy** (*float, optional*) – The modelled contribution to the FRB DM by electrons in the Milky Way. This value is calculated using `calc_dm_galaxy()` Units:  $\text{pc cm}^{-3}$  Default: 0.0
- **dm\_excess** (*float or None, optional*) – The DM excess of the FRB over the estimated Galactic DM. If `dm_excess` is *None*, then `dm_excess` is calculated automatically with `calc_dm_excess()`. Units:  $\text{pc cm}^{-3}$  Default: *None*
- **z\_host** (*float or None, optional*) – The observed redshift of the localised FRB host galaxy. Default: *None*
- **dm\_host\_est** (*float, optional*) – The estimated contribution to the measured FRB DM originating from the FRB's host galaxy. This value is the amount of DM the host galaxy contributes to the observed DM, *not* the DM of the host galaxy. Setting this to a non-zero value and setting `subtract_host=True` when calling `calc_redshift()` accounts for the DM contribution due to the host galaxy. Units:  $\text{pc cm}^{-3}$  Default: 0.0
- **dm\_host\_loc** (*float, optional*) – The dispersion measure of a localised FRB host galaxy. This value is *not* the contribution to the observed DM, but the DM at the host galaxy. The observed DM is `dm_host_loc` but attenuated by a factor of  $(1 + z)$ . To use this, the redshift of the host galaxy must be known. Units:  $\text{pc cm}^{-3}$  Default: 0.0
- **dm\_index** (*float or None, optional*) – The dispersion measure index of the burst  $\alpha$  such that  $\text{DM} \propto \nu^{-\alpha}$  Default: *None*
- **scatt\_index** (*float or None, optional*) – The scattering index ( $\beta$ ) of the FRB pulse. The scattering index describes how the width ( $W$ ) of the FRB pulse evolves with frequency  $\nu$  such that  $W \propto \nu^{-\beta}$ . Default: *None*
- **snr** (*float or None, optional*) – The signal-to-noise ratio of the burst. Default: *None*
- **width** (*float or None, optional*) – The observed width of the pulse obtained by a pulse fitting algorithm. Units: ms Default: *None*
- **peak\_flux** (*float or None, optional*) – The observed peak flux density of the burst. Units: Jy Default: *None*
- **fluence** (*float or None, optional*) – The observed fluence of the FRB. If `fluence` is *None* and both `width` and `peak_flux` are not *None* then `fluence` is automatically calculated with `calc_fluence()`. Units: Jy ms Default: *None*
- **obs\_bandwidth** (*float or None, optional*) – The observing bandwidth. Units: MHz Default: *None*
- **obs\_freq\_central** (*float or None, optional*) – The central observing frequency, Units: MHz Default: *None*
- **utc** (*str or None, optional*) – The UTC time of the FRB Burst. Format should be of the form '1999-01-01T00:00:00.000'. Default: *None*

#### Example

```
>>> import fruitbat
>>> FRB = fruitbat.Frb(879, gl="12:31:40.5", gb="3:41:10.0")
```

(continues on next page)

(continued from previous page)

```
>>> FRB.calc_dm_galaxy()
>>> FRB.calc_redshift()
```

**calc\_comoving\_distance()**

Calculates the comoving distance to the FRB based on its redshift. To calculate a comoving distance either call `calc_redshift()` first to determine a redshift estimate or provide a value for `dm_host`.

**Returns** `astropy.units.Quantity` – The comoving distance to the FRB.

**calc\_dm\_excess()**

Calculates the dispersion measure excess of the FRB by subtracting the DM contribution from the Milky Way.

**Returns** `astropy.units.Quantity` – The dispersion measure excess.

**Notes**

$DM_{\text{excess}}$  is calculated as follows:

$$DM_{\text{excess}} = DM - DM_{\text{galaxy}}$$

**calc\_dm\_galaxy(model='ymw16', include\_halo=False, return\_tau\_sc=False)**

Calculates the dispersion measure contribution of the Milky Way from either (*raj*, *decj*) or (*gl*, *gb*). Uses the YMW16 model of the Milky Way free electron column density.

**Parameters**

- **model** ('ymw16' or 'ne2001', optional) – The Milky Way dispersion measure model. To use 'ne2001' you will need to install the python port. See [https://fruitbat.readthedocs.io/en/latest/user\\_guide/ne2001\\_installation.html](https://fruitbat.readthedocs.io/en/latest/user_guide/ne2001_installation.html) Default: 'ymw16'
- **include\_halo** (bool, optional) – Include the DM of the galactic halo which isn't included in NE2001 or YMW16. This used the YT2020 halo model. Default: False
- **return\_tau\_sc** (bool, optional) – Return the scattering timescale in addition to the DM. Default: False

**Returns**

- **dm\_galaxy** (`astropy.units.Quantity`) – The dispersion measure contribution from the Milky Way of the FRB.
- **tau\_sc** (`astropy.units.Quantity`, optional) – The scattering timescale at 1 GHz (s). Only returns if `return_tau_sc` is *True*.

**calc\_dm\_igm()**

Calculates the dispersion measure of the intergalactic medium along the line-of-sight of the FRB. This can only be done if the redshift and dispersion measure contribution of the FRB host galaxy is known.

**Returns** `astropy.units.Quantity` – The dispersion measure contribution of the IGM.

**Notes**

$DM_{\text{IGM}}$  is calculated as follows:

$$DM_{\text{IGM}} = DM_{\text{excess}} - \frac{DM_{\text{host,loc}}}{1+z}$$

**calc\_energy** (*use\_bandwidth=False*)

Calculates the isotropic energy of the FRB. This is the upper limit to the the true energy since the luminosity distance required is also an upper limit to the true luminosity distance.

**Parameters** *use\_bandwidth* (*bool, optional*) – The default method of calculating the energy of a FRB uses *obs\_freq\_central* as described in Zhang 2018. However some estimates of the FRB energy instead use *obs\_bandwidth* (see Law et al. 2017). Set to *True* to use *obs\_bandwidth* instead of *obs\_freq\_central*. Default: *False*

**Returns** *astropy.units.Quantity* – The estimated isotropic energy of the FRB, in units of ergs.

---

**Notes**

The energy of a FRB is calculated following that in Zhang 2018.

$$E_{FRB} \simeq \frac{4\pi}{1+z} D_L^2 F_{obs} \nu_c$$

Where  $F_{obs}$  is the fluence of the FRB,  $\nu_c$  is the central observing frequency,  $D_L$  is the luminosity distance and  $z$  is the estimated redshift.

In Law et al. 2017, the bandwidth ( $B$ ) is used instead of  $\nu_c$  to estimate energy.

---

**calc\_fluence** ()

Calculates the observed fluence of the FRB. This requires *width* and *peak\_flux* to not be *None*.

**Returns** *astropy.units.Quantity* or *None* – The fluence of the FRB.

---

**Notes**

The fluence ( $F_{obs}$ ) is calculated as follows:

$$F_{obs} = W_{obs} S_{\nu,p}$$

Where  $W_{obs}$  is the with of the FRB pulse and  $S_{\nu,p}$  is the specific peak flux.

---

**calc\_luminosity** (*use\_bandwidth=False*)

Calculates the isotropic peak luminosity of the FRB. This is the upper limit to the the true peak luminosity since the luminosity distance required is also an upper limit to the true luminosity distance.

**Parameters** *use\_bandwidth* (*bool, optional*) – The default method of calculating the luminosity of a FRB uses *obs\_freq\_central* as described in Zhang 2018. However some estimates of the FRB luminosity instead use *obs\_bandwidth* (see Law et al. 2017). Set to *True* to use *obs\_bandwidth* instead of *obs\_freq\_central*. Default: *False*

**Returns** *astropy.units.Quantity* – The estimated isotropic peak luminosity of the FRB in units of ergs/s.

---

**Notes**

The luminosity of a FRB is calculated following that in Zhang 2018.

$$L_{FRB} \simeq 4\pi D_L^2 S_{\nu,p} \nu_c$$

Where  $S_{\nu,p}$  is the specific peak flux of the FRB,  $\nu_c$  is the central observing frequency,  $D_L$  is the luminosity distance.

In Law et al. 2017, the bandwidth ( $B$ ) is used instead of  $\nu_c$  to estimate luminosity.

---

#### `calc_luminosity_distance()`

Calculates the luminosity distance to the FRB based on its redshift. To calculate a luminosity distance either call `calc_redshift()` first to determine a redshift estimate or provide a value for `dm_host`.

**Returns** `astropy.units.Quantity` – The luminosity distance to the FRB.

`calc_redshift` (*method*='Batten2021', *cosmology*='Planck18', *subtract\_host*=False,  
*lookup\_table*=None)

Calculate the redshift of the FRB from its `dm`, `dm_excess` or `dm_excess - dm_host_est`.

#### Parameters

- **method** (*str, optional*) – The dispersion measure-redshift relation to use when calculating the redshift. Available methods: ['Ioka2003', 'Inoue2004', 'Zhang2018', 'Batten2021']. Default: 'Inoue2004'
- **cosmology** (*str, optional*) – The cosmology to assume when calculating the redshift. Available cosmologies: ['WMAP5', 'WMAP7', 'WMAP9', 'Planck13', 'Planck15', 'Planck18', 'EAGLE']. Default: 'Planck18'
- **subtract\_host** (*bool, optional*) – Subtract `dm_host_est` from the `dm_excess` before calculating the redshift. This is used to account for the dispersion measure that arises from the FRB host galaxy. Default: False
- **lookup\_table** (*str or None, optional*) – The path to the lookup table file. If `lookup_table=None` a table will attempted to be loaded from the data directory based on the method name. Default: None

**Returns** *float* – The redshift of the FRB.

---

#### Notes

The `methods` section in the documentation has a description for each of the builtin methods.

The `cosmology` section in the documentation has a list of the cosmological parameters for each cosmology

---

`calc_redshift_conf_int` (*method*='Batten2021', *sigma*=1, *scatter\_percentage*=0,  
*\*\*calc\_redshift\_kwargs*)

Calculates the mean redshift and the confidence interval of an FRB from its `dm`, `dm_excess` or `dm_excess - dm_host_est`.

#### Parameters

- **method** (*str, optional*) – The dispersion measure-redshift relation to use when calculating the redshift. Available methods: %(meth)s. Default: 'Batten2020'
- **cosmology** (*str, optional*) – The cosmology to assume when calculating the redshift. This value is overrided if using a hydrodynamic method. Available cosmologies: %(cosmo)s. Default: 'Planck18'
- **sigma** (*int (1, 2, 3, 4, 5), optional*) – The width of the confidence interval in units of standard deviation. *sigma*=1 is the 68% confidence interval.
- **scatter\_percentage** (*float, optional*) – The amount of line of Default: 0
- **subtract\_host** (*bool, optional*) – Subtract `dm_host_est` from the `dm_excess` before calculating the redshift. This is used to account for the dispersion measure that arises from the FRB host galaxy. Default: False

- **lookup\_table** (*str or None, optional*) – The path to the lookup table file. If `lookup_table=None` a table will attempted to be loaded from the data directory based on the method name. Default: *None*

#### Returns

- *float* – The estimated redshift of the FRB.
- *float* – The redshift confidence interval the FRB.

---

#### Notes

The [methods](#) section in the documentation has a description for each of the builtin methods.

The [cosmology](#) section in the documentation has a list of the cosmological parameters for each cosmology

---

**calc\_redshift\_pdf** (*method='Batten2021', cosmology='Planck18', prior='uniform', subtract\_host=False, lookup\_table=None*)

Calc

**calc\_skycoords** (*frame=None*)

Calculates the skycoord position on the sky of the FRB from (*raj*, *decj*) or (*gl*, *gb*). If both (*raj*, *decj*) and (*gl*, *gb*) are given, preference is given to (*raj*, *decj*).

**Parameters** *frame* (*str or None, optional*) – The type of coordinate frame. If *frame* = *None* then `calc_skycoords()` will use a default frame based on the coordinates given. If *raj* and *decj* are given the default frame is 'icrs'. If *gl* and *gb* are given the default frame is 'galactic'. Default: *None*

**Returns** `astropy.coordinates.SkyCoord` – The sky coordinates of the FRB.

**cosmology**

The cosmology used to calculate redshift.

**Type** *str or None*

**decj**

The declination in J2000 coordinates of the best estimate of the FRB position. This is given in the IRCS frame.

**Type** `astropy.coordinates.Latitude` or *None*

**dm**

The observed dispersion measure of the FRB. This is without Milky Way or host galaxy subtraction.

**Type** `astropy.units.Quantity` or *None*

**dm\_excess**

The dispersion measure with the Milky Way component subtracted. This value approximates the dispersion measure of the intergalactic medium by assuming that the host galaxy contributes zero to the observed dispersion measure. This quantity is calculated using `calc_dm_excess()`.

**Type** `astropy.units.Quantity` or *None*

**dm\_galaxy**

The galactic component of the dispersion measure. This is quantity is calculated using `calc_dm_galaxy()`.

**Type** `astropy.units.Quantity` or *None*

**dm\_galaxy\_model**

The method used to calculate redshift.

**Type** str or None

**dm\_host\_est**

The estimated dispersion measure from the FRB host galaxy.

**Type** `astropy.units.Quantity` or None

**dm\_host\_loc**

The dispersion measure from a localised FRB host galaxy.

**Type** `astropy.units.Quantity` or None

**dm\_igm**

The estimated disperison measure from the IGM. This can be calculated for a FRB for which the redshift of its host galaxy is known. This value is determined using `calc_dm_igm()`.

**Type** `astropy.units.Quantity` or None

**dm\_index**

The dispersion measure index of the burst.

**Type** `astropy.units.Quantity` or None

**fluence**

The observed fluence of the FRB. This is calculated from `calc_fluence()`.

**Type** `astropy.units.Quantity` or None

**gb**

The latitude in galactic coordinates of the best estimate of the FRB position.

**Type** `astropy.coordinates.Latitude` or None

**gl**

The longitude in galactic coordinates of the best estimate of the FRB position.

**Type** `astropy.coordinates.Longitude` or None

**method**

The method used to calculate redshift.

**Type** str or None

**name**

str The name of the FRB object.

**Type** name

**obs\_bandwidth**

The observing bandwidth of the FRB.

**Type** `astropy.units.Quantity` or None

**obs\_freq\_central**

The central observing frequency of the FRB.

**Type** `astropy.units.Quantity` or None

**peak\_flux**

The observed peak flux density of the FRB.

**Type** `astropy.units.Quantity` or None

**plot\_redshift\_pdf** (*\*args*, *\*\*kwargs*)

Plots the redshift pdf and confidence interval for an FRB.

**Parameters**

- **frb** (*fruitbat.Frb*) – An instance of the *fruitbat.Frb* class.
- **method** (*str, optional*) – Default: “Batten2021”
- **sigma** (*[1, 2, 3, 4, 5], optional*) – The width of the confidence interval in units of sigma. Default: 1
- **usetex** (*bool, optional*) – Use LaTeX font when creating the figure. Set this to false to disable Latex fonts. Default: True
- **filename** (*str or None, optional*)

**Returns**

- **fig** (*, optional*)
- **ax** (*, optional*)

**raj**

The right accension in J2000 coordinates of the best estimate of the FRB position. This is given in the IRCS frame.

**Type** *astropy.coordinates.Longitude* or None

**scatt\_index**

The scattering index of the burst.

**Type** *astropy.units.Quantity* or None

**skycoords**

The skycoords of the FRB. This is calculated from either (*raj*, *decj*) or (*gl*, *gb*) using *calc\_skycoords()*.

**Type** *astropy.coordinates.SkyCoord* or None

**snr**

The signal-to-noise ratio of the burst.

**Type** *astropy.units.Quantity*

**utc**

*astropy.time.Time* or None The UTC time of the burst.

**width**

The observed width of the pulse.

**Type** *astropy.units.Quantity* or None

**z**

The estimated redshift of the burst. By default this assumes that the entire *dm\_excess* arrives from the IGM and the host galaxy of the FRB and any surrounding material contribute nothing to the total DM. This should be taken as an upper limit to the bursts true redshift. To provide an estimate of the DM contribution due to he host galaxy, set *dm\_host\_est* to a non-zero value and use *subtract\_host=True* when calling *calc\_redshift()*.

**Type** *astropy.units.Quantity* or None

**z\_conf\_int\_lower**

The lower bound of the redshift confidence interval.

**Type** *astropy.units.Quantity* or None

**z\_conf\_int\_upper**

The upper bound of the redshift confidence interval.

**Type** *astropy.units.Quantity* or None



**z\_host**

The redshift of the localised FRB host galaxy. Note that this an observed quantity, not the estimated redshift  $z$  calculated with `calc_redshift()`.

**Type** `astropy.units.Quantity` or `None`



`fruitbat.methods.ioka2003(z, cosmo, zmin=0)`

Calculates the mean dispersion measure from redshift zero to redshift  $z$  given a cosmology using the Ioka (2003) relation.

**Parameters**

- **z** (*float or int*) – The input redshift.
- **cosmo** (An instance of `astropy.cosmology`) – The cosmology to assume when calculating the dispersion measure at redshift  $z$ .
- **zmin** (*float or int, optional*) – The minimum redshift to begin the integral. This should typically be zero. Default: 0.

**Returns** **dm** (*float*) – The dispersion measure at the redshift  $z$ .

`fruitbat.methods.inoue2004(z, cosmo, zmin=0)`

Calculates the mean dispersion measure from redshift zero to redshift  $z$  given a cosmology using the Inoue (2004) relation.

**Parameters**

- **z** (*float or int*) – The input redshift.
- **cosmo** (An instance of `astropy.cosmology`) – The cosmology to assume when calculating the dispersion measure at redshift  $z$ .
- **zmin** (*float or int, optional*) – The minimum redshift to begin the integral. This should typically be zero. Default: 0.

**Returns** **dm** (*float*) – The dispersion measure at the redshift  $z$ .

`fruitbat.methods.zhang2018(z, cosmo, zmin=0, **kwargs)`

Calculates the mean dispersion measure from redshift zero to redshift  $z$  given a cosmology using the Zhang (2018) relation.

**Parameters**

- **z** (*float or int*) – The input redshift.

- **cosmo** (An instance of `astropy.cosmology`) – The cosmology to assume when calculating the dispersion measure at redshift  $z$ .
- **zmin** (*float or int, optional*) – The minimum redshift to begin the integral. This should typically be zero. Default: 0.

**Keyword Arguments**

- **f\_igm** (*float, optional*) – The fraction of baryons in the intergalactic medium. Default: 0.83
- **free\_elec** (*float, optional*) – The free electron number per baryon in the intergalactic medium. Default: 0.875

**Returns** **dm** (*float*) – The dispersion measure at the redshift  $z$ .

`fruitbat.methods.add_method(name, func)`

Add a user defined method/DM- $z$  relation to the list of available methods.

**Parameters**

- **name** (*str*) – The keyword for the new method.
- **func** (*function*) – The function to calculate the dispersion measure at a given redshift. The first argument of `func` must be  $z$ .

**Returns** *None*

**Example**

```
>>> def simple_dm(z):
    dm = 1200 * z
    return dm
>>> fruitbat.add_method("simple_dm", simple_dm)
```

`fruitbat.methods.available_methods()`

Returns the list containing all the keywords for valid methods.

`fruitbat.methods.reset_methods()`

Resets the list of available methods to the default builtin methods.

`fruitbat.methods.method_functions()`

Returns a dictionary containing the valid method keys and their corresponding dispersion measure functions.

`fruitbat.methods.methods_hydrodynamic()`

Returns a list containing the valid method keys that have used hydrodynamic simulations.

`fruitbat.methods.methods_analytic()`

Returns a list containing the valid method keys that use analytic estimates.

Module for defining different cosmologies

`fruitbat.cosmologies.add_cosmology(name, cosmo)`  
 Adds a user created cosmology to the list of available cosmologies.

### Parameters

- **name** (*str*) – The keyword for the new cosmology.
- **cosmo** (An instance of `astropy.cosmology`) – The cosmology to add to the list of available cosmologies.

### Example

```
>>> params = {"H0": 72.4, "Om0": 0.26}
>>> new_cosmology = fruitbat.cosmology.create_cosmology(parameters=params)
>>> fruitbat.add_cosmology("new_cosmology", new_cosmology)
```

`fruitbat.cosmologies.available_cosmologies()`  
 Returns the list containing all the keywords for valid cosmologies.

`fruitbat.cosmologies.builtin_cosmology_functions()`  
 Returns a dictionary of the builtin cosmologies with keywords and corresponding instances of those cosmologies.

**Returns cosmologies** (*dict*) – Contains the keywords and instances for each cosmology.

`fruitbat.cosmologies.cosmology_functions()`  
 Returns a dictionary containing the valid cosmology keys and the corresponding instance of that cosmology.

`fruitbat.cosmologies.create_cosmology(parameters=None, name=None)`  
 A wrapper to create custom astropy cosmologies.

The only available cosmology types in this method are: `FlatLambdaCDM`, `FlatwCDM`, `LambdaCDM` and `wCDM`.

See [astropy](#) for more details on these types of cosmologies. To create a cosmology of a type that isn't listed above, it will have to be created directly using `astropy.cosmology`.

### Parameters

- **parameters** (*dict or None*) – A dictionary containing the cosmological parameters. The names of the parameters must conform to the same format as the parameters used in the `astropy.cosmology` module. If `parameters = None` then default values for each parameter is used.
- **name** (*str or None, optional*) – The name of the cosmology. Default: *None*

### Returns

- `FlatLambdaCDM` if `flat = True & w = -1`
- `FlatwCDM` if `flat = True & w != -1`
- `LambdaCDM` if `flat = False & w = -1`
- `wCDM` if `flat = False & w != -1`

---

### Notes

Default parameter values:

```
params = {'H0': 70, 'Om0': 0.3, 'Oc0': 0.26, 'Ob0': 0.04,  
          'Neff': 3.04, 'flat': True, 'Tcmb0': 0.0,  
          'm_nu': 0.0, 'w0': -1}
```

If 'flat' is set to False then a value of 'Ode0' (current dark energy density) should be specified.

---

`fruitbat.cosmologies.reset_cosmologies()`

Resets the list of available cosmologies to the default builtin cosmologies.

`fruitbat.cosmologies.Planck13()`

Planck13 instance of FlatLambdaCDM cosmology

(from Planck Collaboration 2014, A&A, 571, A16 (Paper XVI), Table 5 (Planck + WP + highL + BAO))

`fruitbat.cosmologies.Planck15()`

Planck15 instance of FlatLambdaCDM cosmology

(from Planck Collaboration 2016, A&A, 594, A13 (Paper XIII), Table 4 (TT, TE, EE + lowP + lensing + ext))

`fruitbat.cosmologies.Planck18()`

Planck18 instance of FlatLambdaCDM cosmology

(from Planck 2018 results. VI. Cosmological Parameters, A&A, submitted, Table 2 (TT, TE, EE + lowE + lensing + BAO))

`fruitbat.cosmologies.WMAP5()`

WMAP5 instance of FlatLambdaCDM cosmology

(from Komatsu et al. 2009, ApJS, 180, 330, doi: 10.1088/0067-0049/180/2/330. Table 1 (WMAP + BAO + SN ML).)

`fruitbat.cosmologies.WMAP7()`

WMAP7 instance of FlatLambdaCDM cosmology

(from Komatsu et al. 2011, ApJS, 192, 18, doi: 10.1088/0067-0049/192/2/18. Table 1 (WMAP + BAO + H0 ML).)

`fruitbat.cosmologies.WMAP9()`

WMAP9 instance of FlatLambdaCDM cosmology

(from Hinshaw et al. 2013, ApJS, 208, 19, doi: 10.1088/0067-0049/208/2/19. Table 4 (WMAP9 + eCMB + BAO + H0, last column))





```
fruitbat.catalogue.create_analysis_catalogue(filename='fruitbat_analysis_catalogue',
                                             dataset='default', method='Inoue2004',
                                             cosmology='Planck18')
```

Analyses an FRB dataset and produces a CSV file containing the estimated redshift, fluence, energy and luminosity for each FRB in addition to its measured quantities.

### Parameters

- **filename** (*str, optional*) – The output file name. Default: 'fruitbat\_analysis\_catalogue'
- **dataset** (*str, optional*) – The path to the FRBCAT dataset. The dataset is required to have the following columns: 'frb\_name', 'utc', 'telescope', 'rop\_raj', 'rop\_decj', 'rop\_gl', 'rop\_gb', 'rop\_bandwidth', 'rop\_centre\_frequency', 'rmp\_dm', 'rmp\_width', 'rmp\_snr', 'rmp\_flux'. If `dataset='default'` then the builtin dataset will be used. The builtin dataset was last updated: 2019-04-08. Default: 'default'
- **method** (*str, optional*) – The dispersion measure - redshift relation to use when calculating the redshift. Available methods: ['Ioka2003', 'Inoue2004', 'Zhang2018', 'Batten2021']. Default: 'Inoue2004'
- **cosmology** (*str, optional*) – The cosmology to assume when calculating redshift. Available cosmologies: ['WMAP5', 'WMAP7', 'WMAP9', 'Planck13', 'Planck15', 'Planck18', 'EAGLE']. Default: 'Planck18'

### Generates

A csv file with the output of the of the analysis.

```
fruitbat.catalogue.create_methods_catalogue(filename='fruitbat_methods_catalogue',
                                             dataset='default', cosmology='Planck18')
```

Analyses an FRB dataset and produces a CSV file containing the estimated redshift for each method.

### Parameters

- **filename** (*str, optional*) – The output file name. Default: 'fruitbat\_methods\_catalogue'

- **dataset** (*str, optional*) – The path to the FRBCAT dataset. The dataset is required to have the following columns: ‘frb\_name’, ‘utc’, ‘telescope’, ‘rop\_raj’, ‘rop\_decj’, ‘rop\_gl’, ‘rop\_gb’, ‘rmp\_dm’. If `dataset='default'` then the builtin dataset will be used. The builtin dataset was last updated: 2019-04-08. Default: ‘default’
- **cosmology** (*str, optional*) – The cosmology to assume when calculating redshift. Available cosmologies: [‘WMAP5’, ‘WMAP7’, ‘WMAP9’, ‘Planck13’, ‘Planck15’, ‘Planck18’, ‘EAGLE’]. Default: ‘Planck18’

## Generates

A csv file with the output of the of the analysis.

`fruitbat.catalogue.read_frb_row(row)`

Reads the row of a DataFrame and retrieves the data in the correct format.

**Parameters** `row` (*Series*) – The series containing FRB data.

**Returns** *A dictionary containing the FRB paramemters*

## 11.1 table

```
fruitbat.table.create(method, output_dir='data', filename=None, zmin=0, zmax=20,
                      num_samples=10000, **method_kwargs)
```

Creates lookup table

which can be read

in using `load()`.

### Parameters

- **method** (*str*) – The DM-redshift relation to assume when creating the table.
- **output\_dir** (*str, optional*) – The path of the output directory. If `output_dir = 'data'`, then created table will be created in the same directory with the builtin tables and will be found when using functions such as `calc_redshift()`. Default: 'data'
- **filename** (*str, optional*) – The output filename. If `name=None` then the filename will become `custom_method`. Default: *None*
- **zmin** (*int or float, optional*) – The minimum redshift in the table. Default: 0
- **zmax** (*int or float, optional*) – The maximum redshift in the table. Default: 20
- **num\_samples** (*int, optional*) – The number of sample dispersion measure and redshifts. Default: 10000

### Keyword Arguments

- **cosmo** (An instance of `astropy.cosmology`, *optional*) – The cosmology to assume when calculating the outputs of the table. Required when creating new tables of 'Ioka2003', 'Inoue2004', 'Zhang2018'.
- **free\_elec** (*float or int, optional*) – The amount of free electrons per proton mass in the Universe. This applies when using 'Zhang2018'. Must be between 0 and 1. Default: 0.875.

- **f\_igm** (*float or int, optional*) – The fraction of baryons in the intergalactic medium. This applies when using 'Zhang2018'. Must be between 0 and 1. Default: 0.83

**Returns** *string* – The path to the generated hdf5 file containing the table data.

### Generates

A hdf5 file containing datasets for 'DM' and 'z'.

### Example

```
>>> def simple_dm(z):  
    dm = 1200 * z  
    return dm  
>>> fruitbat.add_method("simple_dm", simple_dm)  
>>> fruitbat.table.create("simple_dm")  
>>> frb = fruitbat.Frb(1200)  
>>> frb.calc_redshift(method="simple_dm")  
<Quantity 1.>
```

`fruitbat.table.get_z_from_table(dm, table, cosmology=None)`

Calculates the redshift from a dispersion measure by interpolating a lookup table

#### Parameters

- **dm** (*float*) – The input dispersion measure
- **table** (`numpy.lib.npyio.NpzFile`) – The lookup table with 'dm' and 'z' arrays.

**Returns** *z (float)* – The redshift corresponding to the input disperison measure.

### Example

```
>>> table = fruitbat.table.load('Zhang2018_Planck18.npz')  
>>> fruitbat.get_z_from_table(1000, table)  
1.1087964578507539
```

`fruitbat.table.get_table_path(filename, datadir='data')`

#### Parameters

- **filename**
- **datadir**

**Returns** *path (string)* – The path to the data file

`fruitbat.plot._fruitbat_colors()`

The official list of fruitbat plotting colours.

`fruitbat.plot.cosmology_comparison(filename="", extension='png', usetex=False, passed_ax=None, **kwargs)`

Create a plot comparing how the estimated redshift changes as a function of dispersion measure for each cosmology.

### Parameters

- **filename** (*string, optional*)
- **The filename of the saved figure. Default** (“*cosmology\_comparison*”)
- **extension** (*string, optional*) – The format to save the figure. e.g “png”, “pdf”, “eps”, etc... Default: “png”

### Generates

A figure displaying how estimated redshift changes as a function of dispersion measure for each of the different cosmologies.

`fruitbat.plot.method_comparison(filename=None, extension='png', usetex=False, passed_ax=None, **kwargs)`

Create a plot comparing how estimated redshift changes as a function of dispersion measure for each DM-z relation.

### Parameters

- **filename** (*string or None, optional*) – The filename of the saved figure. Default: *None*
- **extension** (*string, optional*) – The format to save the figure. e.g “png”, “pdf”, “eps”, etc... Default: “png”
- **usetex** (*bool, optional*) – Use LaTeX for fonts.
- **passed\_ax** (*or None, optional*)

## Generates

A figure displaying how estimated redshift changes as a function of dispersion measure for each of the different cosmologies.

```
fruitbat.plot.redshift_pdf(frb, method='Batten2021', sigma=1, usetex=True, filename=None,  
                           outputdir=None)
```

Plots the redshift pdf and confidence interval for an FRB.

### Parameters

- **frb** (*fruitbat.Frb*) – An instance of the *fruitbat.Frb* class.
- **method** (*str*, *optional*) – Default: “Batten2021”
- **sigma** (*[1, 2, 3, 4, 5]*, *optional*) – The width of the confidence interval in units of sigma. Default: 1
- **usetex** (*bool*, *optional*) – Use LaTeX font when creating the figure. Set this to false to disable Latex fonts. Default: True
- **filename** (*str or None*, *optional*)

### Returns

- **fig** (*, optional*)
- **ax** (*, optional*)

```
fruitbat.plot.set_rc_params(usetex=False)
```

Set the rcParams that will be used in all the plots.

The collection of utility functions for Fruitbat.

`fruitbat.utils.check_keys_in_dict(dictionary, keys)`

Checks that a list of keys exist in a dictionary.

**Parameters**

- **dictionary** (*dict*) – The input dictionary.
- **keys** (*list of strings*) – The keys that the dictionary must contain.

**Returns** *bool* – Returns *True* if all required keys exist in the dictionary. Otherwise a *KeyError* is raised.

`fruitbat.utils.check_type(value_name, value, dtype, desire=True)`

Checks the type of a variable and raises an error if not the desired type.

**Parameters**

- **value\_name** (*str*) – The name of the variable that will be printed in the error message.
- **value** – The value of the variable
- **dtype** (*dtype*) – The data type to compare with *isinstance*
- **desire** (*boolean, optional*) – If *desire = True*, then the error will be raised if value does not have a data type of *dtype*. If *desire = False*, then the error will be raised if value does have a data type of *dtype*.

**Returns** *None*

`fruitbat.utils.calc_mean_from_pdf(x, pdf, dx=None)`

Calculates the mean of a probability density function

**Parameters**

- **x** (*np.ndarray*) – The x values.
- **pdf** (*np.ndarray*) – The value of the PDF at x.

- **dx** (*np.ndarray or None, optional*) – The spacing between the x bins. If *None*, then the bins are assumed to be linearly spaced.

**Returns** **mean** (*float*) – The mean of the PDF.

`fruitbat.utils.calc_median_from_pdf(x, pdf)`

Calculates the median of a PDF.

**Parameters**

- **x** (*np.ndarray*) – The x values.
- **pdf** (*np.ndarray*) – The value of the PDF at x.

**Returns** **median** (*float*) – The median of the PDF.

`fruitbat.utils.calc_std_from_pdf(x, pdf, dx=None)`

Calculates the standard deviation from a probability density function.

**Parameters**

- **x** (*np.ndarray*) – The x values.
- **pdf** (*np.ndarray*) – The value of the PDF at x.
- **dx** (*np.ndarray or None, optional*) – The spacing between the x bins. If *None*, then the bins are assumed to be linearly spaced.

**Returns** **std** (*float*) – The standard deviation of the PDF.

`fruitbat.utils.calc_variance_from_pdf(x, pdf, dx=None)`

Calculates the variance from a probability density function.

**Parameters**

- **x** (*np.ndarray*) – The x values.
- **pdf** (*np.ndarray*) – The value of the PDF at x.
- **dx** (*np.ndarray or None, optional*) – The spacing between the x bins. If *None*, then the bins are assumed to be linearly spaced.

**Returns** **variance** (*float*) – The variance of the PDF.

`fruitbat.utils.calc_z_from_pdf_percentile(x, pdf, percentile)`

**Parameters**

- **x** (*np.ndarray*) – The x values of the PDF.
- **pdf** (*np.ndarray*) – The value of the PDF at x.
- **percentile** (*float*) – The percentile of the PDF.

**Returns** **redshift** (*float*) – The redshift at the given percentile.

`fruitbat.utils.normalise_to_pdf(hist, bin_widths)`

`fruitbat.utils.redshift_prior(zbins, prior='uniform')`

`fruitbat.utils.sigma_to_pdf_percentiles(sigma)`

Looks up the percentile range of Gaussian for a given standard deviation.

**Parameters** **sigma** (*[1, 2, 3, 4, 5]*) – The standard deviation to calculate a percentile.

**Returns**

- **Lower** (*float*) – The lower percentile



- **Higher** (*float*) – The higher percentile

### Example

```
>>> sigma_to_pdf_percentiles(1)
(0.158655254, 0.841344746)
```



### f

- `fruitbat.catalogue`, 37
- `fruitbat.cosmologies`, 33
- `fruitbat.methods`, 31
- `fruitbat.plot`, 41
- `fruitbat.table`, 39
- `fruitbat.utils`, 43



## Symbols

`_fruitbat_colors()` (in module `fruitbat.plot`), 41

## A

`add_cosmology()` (in module `fruitbat.cosmologies`), 33

`add_method()` (in module `fruitbat.methods`), 32

`available_cosmologies()` (in module `fruitbat.cosmologies`), 33

`available_methods()` (in module `fruitbat.methods`), 32

## B

`builtin_cosmology_functions()` (in module `fruitbat.cosmologies`), 33

## C

`calc_comoving_distance()` (`fruitbat.Frb` method), 23

`calc_dm_excess()` (`fruitbat.Frb` method), 23

`calc_dm_galaxy()` (`fruitbat.Frb` method), 23

`calc_dm_igm()` (`fruitbat.Frb` method), 23

`calc_energy()` (`fruitbat.Frb` method), 23

`calc_fluence()` (`fruitbat.Frb` method), 24

`calc_luminosity()` (`fruitbat.Frb` method), 24

`calc_luminosity_distance()` (`fruitbat.Frb` method), 25

`calc_mean_from_pdf()` (in module `fruitbat.utils`), 43

`calc_median_from_pdf()` (in module `fruitbat.utils`), 44

`calc_redshift()` (`fruitbat.Frb` method), 25

`calc_redshift_conf_int()` (`fruitbat.Frb` method), 25

`calc_redshift_pdf()` (`fruitbat.Frb` method), 26

`calc_skycoords()` (`fruitbat.Frb` method), 26

`calc_std_from_pdf()` (in module `fruitbat.utils`), 44

`calc_variance_from_pdf()` (in module `fruitbat.utils`), 44

`calc_z_from_pdf_percentile()` (in module `fruitbat.utils`), 44

`check_keys_in_dict()` (in module `fruitbat.utils`), 43

`check_type()` (in module `fruitbat.utils`), 43

`cosmology` (`fruitbat.Frb` attribute), 26

`cosmology_comparison()` (in module `fruitbat.plot`), 41

`cosmology_functions()` (in module `fruitbat.cosmologies`), 33

`create()` (in module `fruitbat.table`), 39

`create_analysis_catalogue()` (in module `fruitbat.catalogue`), 37

`create_cosmology()` (in module `fruitbat.cosmologies`), 33

`create_methods_catalogue()` (in module `fruitbat.catalogue`), 37

## D

`decj` (`fruitbat.Frb` attribute), 26

`dm` (`fruitbat.Frb` attribute), 26

`dm_excess` (`fruitbat.Frb` attribute), 26

`dm_galaxy` (`fruitbat.Frb` attribute), 26

`dm_galaxy_model` (`fruitbat.Frb` attribute), 26

`dm_host_est` (`fruitbat.Frb` attribute), 27

`dm_host_loc` (`fruitbat.Frb` attribute), 27

`dm_igm` (`fruitbat.Frb` attribute), 27

`dm_index` (`fruitbat.Frb` attribute), 27

## F

`fluence` (`fruitbat.Frb` attribute), 27

`Frb` (class in `fruitbat`), 21

`fruitbat.catalogue` (module), 37

`fruitbat.cosmologies` (module), 33

`fruitbat.methods` (module), 31

`fruitbat.plot` (module), 41

`fruitbat.table` (module), 39

`fruitbat.utils` (module), 43

## G

`gb` (*fruitbat.Frb attribute*), 27  
`get_table_path()` (*in module fruitbat.table*), 40  
`get_z_from_table()` (*in module fruitbat.table*), 40  
`gl` (*fruitbat.Frb attribute*), 27

## I

`inoue2004()` (*in module fruitbat.methods*), 31  
`ioka2003()` (*in module fruitbat.methods*), 31

## M

`method` (*fruitbat.Frb attribute*), 27  
`method_comparison()` (*in module fruitbat.plot*), 41  
`method_functions()` (*in module fruitbat.methods*), 32  
`methods_analytic()` (*in module fruitbat.methods*), 32  
`methods_hydrodynamic()` (*in module fruitbat.methods*), 32

## N

`name` (*fruitbat.Frb attribute*), 27  
`normalise_to_pdf()` (*in module fruitbat.utils*), 44

## O

`obs_bandwidth` (*fruitbat.Frb attribute*), 27  
`obs_freq_central` (*fruitbat.Frb attribute*), 27

## P

`peak_flux` (*fruitbat.Frb attribute*), 27  
`Planck13()` (*in module fruitbat.cosmologies*), 34  
`Planck15()` (*in module fruitbat.cosmologies*), 34  
`Planck18()` (*in module fruitbat.cosmologies*), 34  
`plot_redshift_pdf()` (*fruitbat.Frb method*), 27

## R

`raj` (*fruitbat.Frb attribute*), 28  
`read_frb_row()` (*in module fruitbat.catalogue*), 38  
`redshift_pdf()` (*in module fruitbat.plot*), 42  
`redshift_prior()` (*in module fruitbat.utils*), 44  
`reset_cosmologies()` (*in module fruitbat.cosmologies*), 34  
`reset_methods()` (*in module fruitbat.methods*), 32

## S

`scatt_index` (*fruitbat.Frb attribute*), 28  
`set_rc_params()` (*in module fruitbat.plot*), 42  
`sigma_to_pdf_percentiles()` (*in module fruitbat.utils*), 44  
`skycoords` (*fruitbat.Frb attribute*), 28  
`snr` (*fruitbat.Frb attribute*), 28

## U

`utc` (*fruitbat.Frb attribute*), 28

## W

`width` (*fruitbat.Frb attribute*), 28  
`WMAP5()` (*in module fruitbat.cosmologies*), 34  
`WMAP7()` (*in module fruitbat.cosmologies*), 34  
`WMAP9()` (*in module fruitbat.cosmologies*), 34

## Z

`z` (*fruitbat.Frb attribute*), 28  
`z_conf_int_lower` (*fruitbat.Frb attribute*), 28  
`z_conf_int_upper` (*fruitbat.Frb attribute*), 28  
`z_host` (*fruitbat.Frb attribute*), 28  
`zhang2018()` (*in module fruitbat.methods*), 31