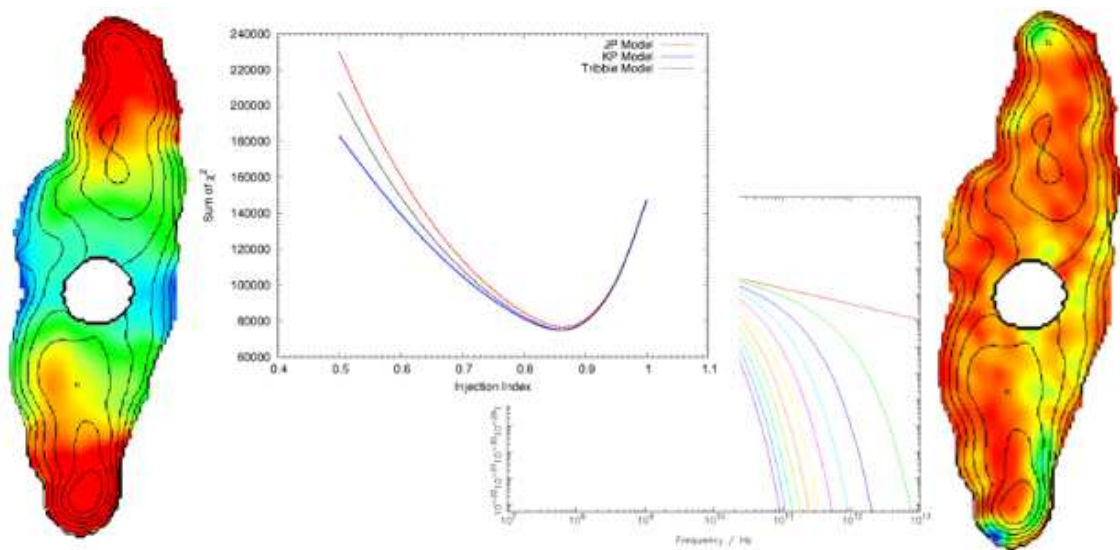# Broadband Radio Astronomy Tools (BRATS) Cookbook

Developed by Jeremy J. Harwood

jeremy.harwood@physics.org

If you have made use of this software, please reference
Harwood et al., 2013, MNRAS, 435, 3353

For the latest updates and support visit http://www.askanastronomer.co.uk/brats

Updated for version 2.4.0 on 29th January 2015

Contents

1

Introduction

1.1 An introduction to the BRATS software package

The broad-bandwidth capabilities of next generation telescopes such as the JVLA, LOFAR e-MERLIN and, ultimately, the SKA mean that the spectrum of any given source varies significantly within the bandwidth of any given observation. Detailed spectral analysis taking this variation into account is set to become standard practice when dealing with any new broadband radio observations but, as with any new technology, there is often a lack of software and appropriate methods of analysis to use these new data to their maximum potential. The Broadband Radio Astronomy ToolS software package (BRATS) was therefore developed to overcome this problem.

Written in C, BRATS is a terminal-based Linux software package which provides a wide range of model fitting, visualization and statistical tools for the analysis of radio maps with good frequency coverage over a large frequency range. This is most often applicable to broadband radio maps where multiple images can be produced within a single pointing, but can also be used for simple analysis of narrowband observations or where a large number of narrowband images are available. The primary function of the software is to provide a set of analysis tools for determining the spectral profile of radio sources, with a particular emphasis on models of spectral ageing.

1.2 Obtaining and installing BRATS

BRATS has successfully been installed on a range of Linux desktop distributions (Debian, Fedora, Mint), as well as Mac OSX and in stand-alone server and cluster environments and can be downloaded directly

from the website[1]. The majority of prerequisites come as standard in most Linux distributions (e.g. a C compiler such as GCC) but is dependent on two non-standard libraries, both of which must first be obtained before installation; PGPLOT[2] and FUNTOOLS[3]. In general, using standard repository tools (e.g. yum or apt-get) to obtain these libraries causes the least compatibility issues and provides the easiest method of installation.

The BRATS tar file should next be extracted (the location here is not important but should be easy to locate), and the Makefile edited to point to the dependencies and the desired install location. Environment variables pointing to the PGPLOT libraries should then be set, for example

```
setenv PGPLOT_DIR /usr/local/pgplot/      #csh
export PGPLOT_DIR=/usr/local/pgplot/      #bash
```

Depending on your setup, you may also be required to add one or more of the dependancies e.g. for PGPLOT to your shared libraries search path

```
setenv LD_LIBRARY_PATH /usr/local/pgplot/      #csh
export LD_LIBRARY_PATH=/usr/local/pgplot/      #bash
```

If your system already has values assigned to LD_LIBRARY_PATH then `:$LD_LIBRARY_PATH` should be added to the end of the command.

These paths are required whenever BRATS is run and it is therefore recommend that these environment variables are entered these into a shell profile (e.g. .cshrc, .bashrc). These instructions along with example installation parameters are detailed within the Makefile for ease of reference.

Once completed, BRATS can then be installed by running the '*make brats*' command. At this point it is recommended that an alias is made to the BRATS executable, for example

```
alias brats '/soft/brats/brats'      #csh
alias brats='/soft/brats/brats'      #bash
```

The software should now be installed and can be accessed by typing '*brats*' in a terminal window.

Before analysis of a data set can begin, BRATS also requires two directories to be created. By default, these are assumed to be named *images* and *data* and to be located in the folder from which the software was launched (the working directory). It is recommended that the working directory also contains the region files and subfolders containing the radio maps, but more complex setups can be also be achieved through use of the '*dataloc*' and '*imageloc*' commands. The recommended structure for standard use is shown in Figure 1.1.

---

[1]http://www.askanastronomer.co.uk/brats/
[2]http://www.astro.caltech.edu/%7Etjp/pgplot/
[3]https://www.cfa.harvard.edu/%7Ejohn/funtools/
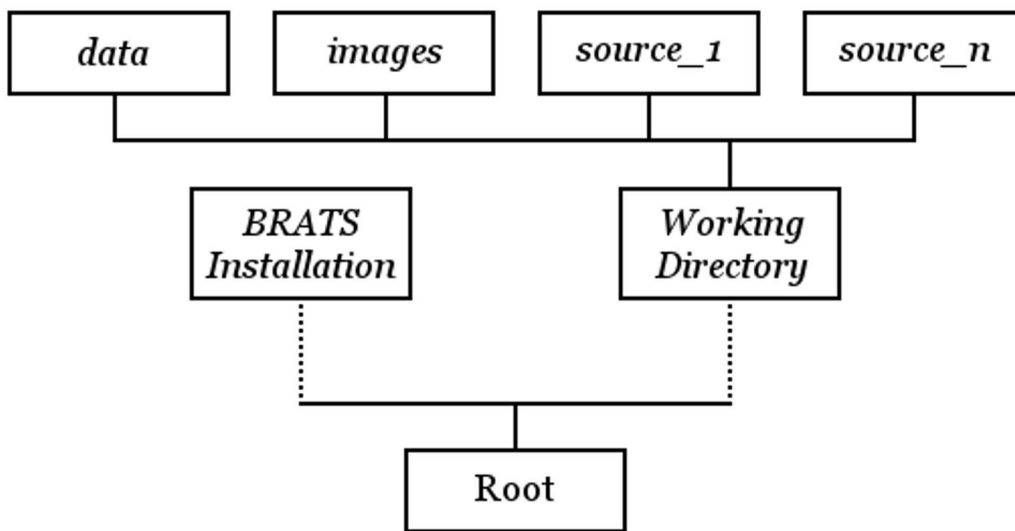
Figure 1.1: Recommended directory structure for BRATS usage. Dotted lines represent the users preferred path, solid lines indicate directories located within the preceding level. Region files should ideally be contained within the working directory. Source directories should contain only the FITS files to be loaded.

2

Quick start tutorial

This quick start guide is designed to get the standard user up and running with BRATS and explain usage of some of its main features. The main focus of this document is the fitting of spectral ageing models to a data set and determination of some of the key parameters. More extensive details of all the features in BRATS can be found in the detailed usage section of the cookbook.

2.1   Setup and loading data
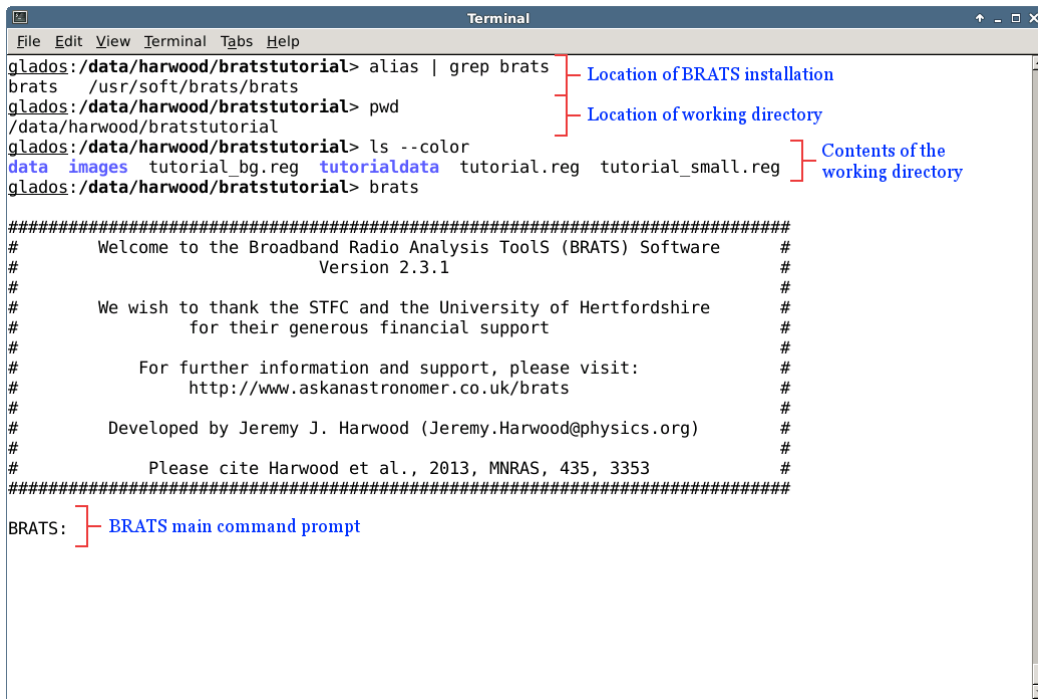
It is assumed that BRATS has already been installed on your system and a working directory has been created containing the recommended images and data folders as per Section 1.2 (Figures 1.1 and 2.1). For this tutorial, we will be using a simulated data set of a (single injection) shock front with ageing plasma downstream which can be downloaded from:

`http://www.askanastronomer.co.uk/brats/downloads/tutorialdata.tar.gz`

This file consists of (simulated) radio maps contained in their own folder and 3 region files; a background region and 2 source region files. Here, we will use the region file which loosely encompasses the entire source (Figure 2.2); however, this can be very time consuming when running model fits. A second region file which only includes a subsection of the source is therefore included. If you are short of time (or computing power!) then simply substitute 'tutorial.reg' for 'tutorial_small.reg' during the load phase. This may cause some of the values (e.g. sum of χ^2) to differ from those described within the tutorial, but the underlying principles remain unchanged. Details of the source are given in Table 2.1.

Once the data and regions files have been downloaded and extracted to your working directory, the first step is to set our initial source detection limit to 5σ via the '*sigma*' parameter. The data set can then be loaded into BRATS using the '*load*' command and entering the data folder, background region,

6

Figure 2.1: An example directory structure for standard BRATS usage.

region file and redshift when promoted. Note that BRATS will run commands 'inline' i.e. one after the other, hence can be copied and pasted directly from the tutorial if desired. This is also a useful feature for keeping track of commands or saving time when fitting real data sets by storing them in a standard text file which can then be run to reproduce results at a later date.

```
sigma
5
load
tutorialdata
tutorial_bg.reg
tutorial.reg
0.1954
```

A variety of statistics should be displayed on the screen about the maps which have been loaded, such as the RMS values, and a flux map of one of the images displayed in a separate window. Note the warning stating that the data style could not be automatically determined due the reduction format not being explicitly stated in the header. In such cases, BRATS reverts to a default format which in this case
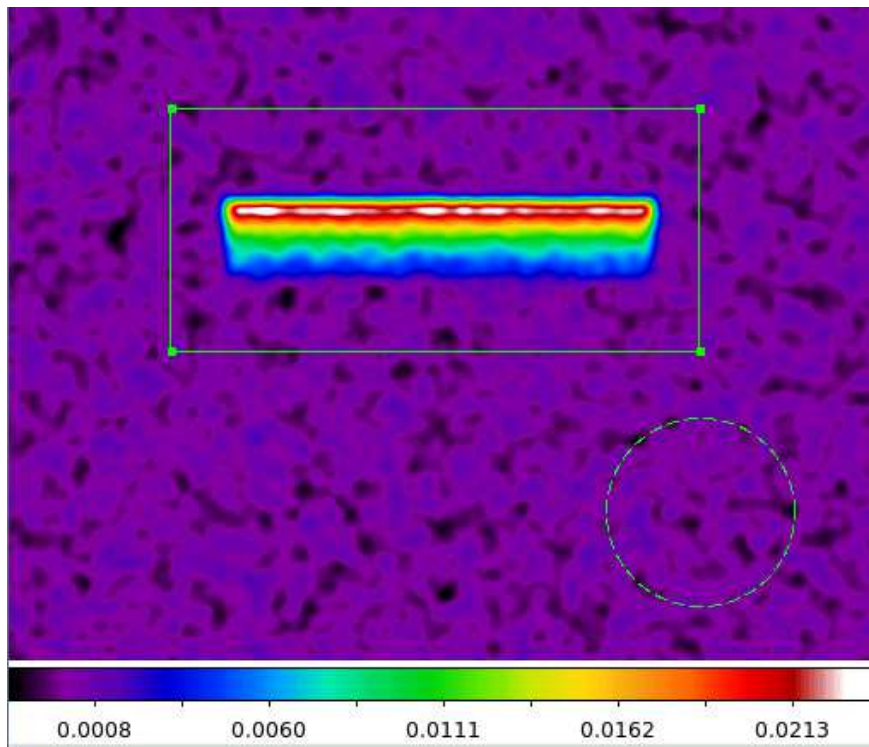
Figure 2.2: Initial source selection regions used in this tutorial. The solid green line shows the main region loosely encompassing the entire source (tutorial.reg) and the dashed green line the background region (tutorial_bg.reg) used for calculating the off-source noise.

matches the CASA format of our data. If our data had been instead made in AIPS and was missing this header information an error would have been thrown and we would have needed to switch the default format to AIPS using the '*casadata*' command.

Note that when loading real data into BRATS a quirk in the FUNTOOLS library used to read the FITS files means there is an upper limit on the size of the image header. If exceeded, BRATS will exit with the error 'no WCS information in file while parsing filter at: XX:XX:XX.XXX'. This normally occurs due to a large history accumulated during reduction and can be fixed by running the AIPS Stalin command or by setting HISTORY = False when exporting images from CASA.

2.2 Setting the regions

Now that the data has been loaded, we must set the regions to which we will be fitting the models. A variety of different region selection options are available but here we will be applying the most widely

Table 2.1: Simulated tutorial source details.

Property	Value	Units
Image size	512 x 512	pixels
Simulated cell size	1	arcsec
Convolved beam size	8	arcsec
Maximum age	40	myears
Injection index	0.77	
Redshift	0.1954	

used method of fitting on a pixel by pixel basis via the adaptive regions command ('*setregions*'). To do this we set the maximum search area ('*searcharea*') and signal to noise ratio ('*signaltonoise*') to 1 to enable the selection of regions which are a single pixel in size (these are the current default settings, but are included it here for illustrative purposes). By default, the standard flux calibration errors for the JVLA are assumed for calculating the errors of each region; however, for these simulated data the errors are much lower. We therefore set the flux calibration errors for all maps to 1 per cent by setting the '*fluxcalerrors*' parameter. We also set the on source noise multiplier to 3 via the '*onsource*' parameter. This factor accounts for the increase in thermal noise within the source compared to the relatively simple to model empty region where the off-source RMS is measured. Further details of this parameters, along with a range of other region selection options and parameters, are given in Section 3.3 and by Harwood et al. (2013).

```
searcharea
1
signaltonoise
1
fluxcalerror
0
888
0.01
onsource
3
setregions
0
```
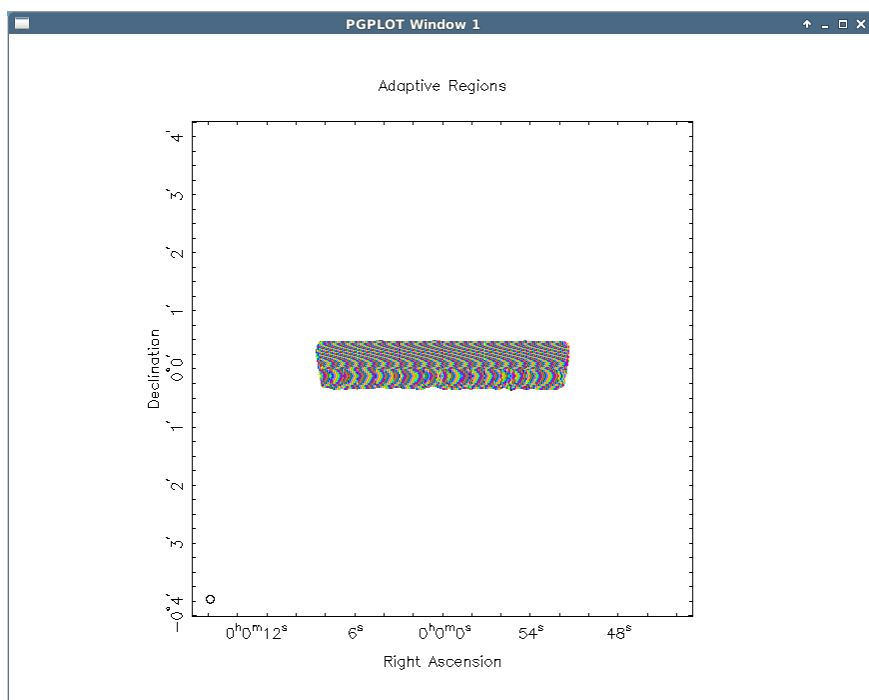
Figure 2.3: Adaptive region results for the simulated tutorial source. Note that the 'wave' pattern is a result of a limited number of colours used in the mapping, rather than having any physical meaning.

Once run, the total number of regions created should be output to the terminal and the regions mapped in a separate window (Figure 2.3). Note that only a limited amount of colours are used for this mapping and so patterns can sometimes appear due to the colours cycling, particularly when setting a large number of single pixel regions.

2.2.1 Checking the data

At this point it is usually a good idea to check the data to ensure everything is as expected and detect any potential problems before fitting the more time consuming spectral ageing models. One way to do this is to perform spectral index fitting ('*specindex*') on the data set which can be particularly useful in flagging up any alignment issues which may exist between the individual image. The type of spectral index fitting performed can be selected using the '*specindexcalctype*' parameter but here we select the default weighted least squares method.

```
specindex
0
```

10

Figure 2.4: Spectral index map of the simulated tutorial source.

You should see the minimum and maximum spectral index output to the terminal along with the spectral index mapped as a function of position in a separate window (Figure 2.4). We see no obvious signs of problems with these data or the regions selected e.g. unexpected sharp gradients around the edge of the source which may suggest alignment issues, and so can be confident that the data are now ready to be fitted with the more complex spectral ageing models.

For data sets where problems do exist, plotting of the flux as a function of frequency with the spectral index fits overlaid using the '*plotspecindex*' can often prove useful in tracking down any problems. In most cases, one does not wish to view the plot of every region fitted (e.g. here we would have almost 10,000 plots!), hence a subset of the data can be selected using the '*skip*' parameter (this parameter applies to all plotting of this type).

2.3   (Optional) Finding the injection index

One of the key parameters of spectral ageing models is the injection index which describes the initial distribution of the electron population. Classically, this has been found by taking the spectral index at the assumed site of particle acceleration; however, this is normally limited to small spatial regions which

11
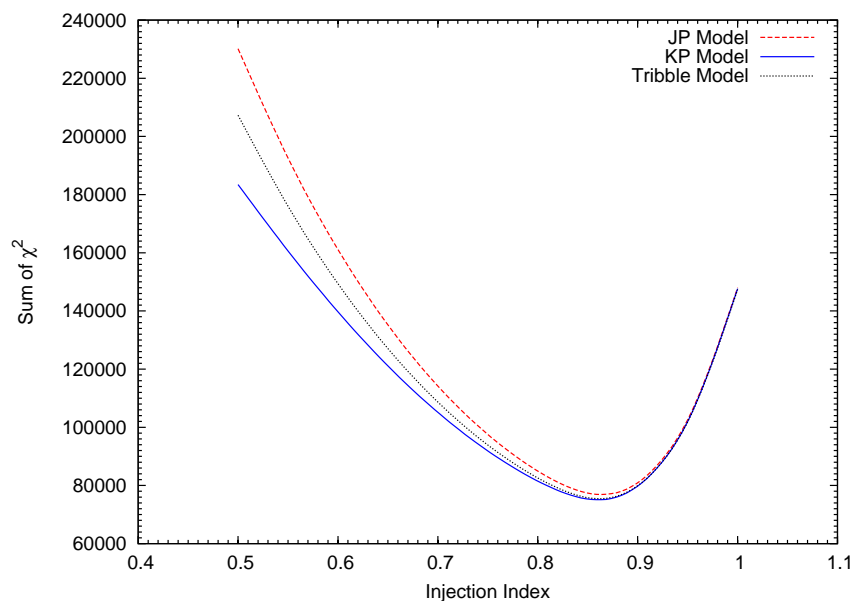
Figure 2.5: Injection index χ^2 minimisation curve for the FR-II galaxy 3C436 taken from Harwood et al. (2013).

may be influenced by, for example, convolution effects (e.g. Stroe et al., 2014). Indeed, looking back at the spectral index map created in Section 2.2.1 we see that for the full source, the minimum spectral index is only 0.75, lower than the original injection index of 0.77 used in the simulation of the source. Conversely, using the small region file results in a minimum spectral index of 0.78 as we lie in a region where a small, ~1 pixel deep section of the acceleration region has been excluded during the region selection process. Although these shifts are small in our example, the effects can be more prominent when dealing with real, often much more messy, data sets.

BRATS therefore includes the '*findinject*' command which sets the injection index as a pseudo-free variable and minimises the χ^2 value of the model fits, allowing all of the source to be used in its determination (Harwood et al., 2013).

Note that this process is very time intensive and therefore recommend using the small region file, rather than the full source for the purposes of this tutorial.

To find the injection index of the source, we must first setup the model fitting parameters (more on these later, but for now we will just use the values stated below). First, we must set a reasonable range of spectral ages to be fitted. As these are simulated data, we know the oldest possible age of the source so, allowing the search to go a reasonable amount above this value, we set the '*myears*' parameter to 50 megayears. The magnetic field strength (in units of Tesla) must also be set via the '*bfield*' parameter to

0.6e-9 T. We must also set a search range for the injection index and the number of intervals between these two values using the '*maxinject*', '*mininject*' and '*injectintervals*' command respectively. The number of fits performed goes as '*injectintervals*'+1 e.g. here for time considerations we keep the search range narrow and set '*maxinject*' to 0.78, '*mininject*' to 0.76 and '*injectintervals*' to 2, which will produce 3 data points at injection indices of 0.76, 0.77 and 0.78.

Once these parameters have been set, we can run the '*findinject*' command selecting JP as our chosen ageing model. Note that all parameters which affect standard spectral age fitting also influence the '*findinject*' command. Some of these parameters are discussed later on in this tutorial and full details can be found in Section 3.4.5.

```
myears
50
bfield
0.6e-9
maxinject
0.78
mininject
0.76
injectintervals
2
findinject
0
1
```

Using the small region file on a standard work station this process should take around 1-2 hours to complete (around 1 day for the full source), but is highly dependant on the number and speed of the CPUs available. The resulting χ^2 values can be exported using the '*exportdata*' command which we look at in further detail in Section 2.6. This is useful for plotting the χ^2 curve as a function of injection index and determining the associated error (see Section 3.4.5). An example of such a plot for a real data set is shown in Figure 2.5.

For real sources, we recommend first using a broad search with reasonably large intervals to narrow down the likely injection index range, then repeating the steps at your required precision. Where a cluster or multiple machines are available it is often a good idea to break the search into sections and run separate instances of BRATS e.g. node 1 searches from 0.5 to 0.7, node 2 searches from 0.75 to 1.0 and so on.

```
Model can be rejected with 68 per cent confidence at X^2 > 5.86 (Reduced: 1.17)
Model can be rejected with 90 per cent confidence X^2 > 9.24 (Reduced: 1.85)
Model can be rejected with 95 per cent confidence X^2 > 11.07 (Reduced: 2.21)
Model can be rejected with 99 per cent confidence X^2 > 15.09 (Reduced: 3.02)
Model can be rejected with 99.5 per cent confidence X^2 > 16.75 (Reduced: 3.35)
Model can be rejected with 99.9 per cent confidence X^2 > 20.52 (Reduced: 4.10)
Model can be rejected with 99.99 per cent confidence X^2 > 20.52 (Reduced: 5.15)

Sum of X^2 6043.55
Average X^2 0.62
Average normalisation 3.37e+15
Average Reduced X^2 0.12 for 5 degrees of freedom
Minimum Age: 0.00 +6.12 -0.00 Megayears (Region 8683)
Maximum Age: 40.97 +2.37 -2.90 Megayears (Region 57)

Bin Statistics:
=====================================================

 Index  |   Confidence Level  |   Regions  |    Fraction

   0    |     < 68 per cent   |    9715    |      1.00
   1    |    68 - 90 per cent |     0      |      0.00
   2    |    90 - 95 per cent |     0      |      0.00
   3    |    95 - 99 per cent |     0      |      0.00
   4    |     > 99 per cent   |     0      |      0.00

=====================================================

Model cannot be rejected at the 68 per cent significant level based on median binning

BRATS: BRATS:
```

Figure 2.6: Model fit results for the full tutorial source using the JP model of spectral ageing.

This is particularly useful for the Tribble model which can take long periods of time when fitting to large sources.

2.4   Model fitting

One of the main functions of BRATS is the fitting of spectral ageing models. A variety of different models and parameters detailed in Section 3.4.2 are available for use, but here we focus on fitting the single injection JP model to the pixel by pixel regions we previously created.

We begin by setting the injection index to the value of 0.77 determined in the previous section via the '*injectionindex*' parameter. For the benefit of those who skipped the previous optional section, we set the maximum spectral age to search range via the '*myears*' parameter to 50 megayears and the magnetic field strength (in units of Tesla) to 0.6e-9 T using the '*bfield*' parameter.

The accuracy to which BRATS will attempt to determine the spectral age of the regions is determined by two key parameter; '*ageres*' and '*levels*'. The '*levels*' parameter sets how "deep" in terms of age precision the model fitting should attempt to go, where each level is of an ever decreasing range of ages. '*ageres*' sets the resolution of the ages attempted for each level. For example, if we fit a model between

0 and 10 megayears, with '*levels*' set to 3 and '*ageres*' set to 10, the first level of fitting will be between 0 and 10 megayears with a step size 1 megayear. If the minimum χ^2 is the found to be at 5 megayears, fitting will be performed between 4.5 and 5.5 megayears in steps of 0.1 and so on until the number of levels required is reached.

For testing and initial fitting runs (e.g. to check the data quality), 3 levels are recommended, as increasing the precision also increases the time required for the model fitting. For final fitting and science purposes it is normally best to increase this to 5 levels. For time considerations we set '*levels*' to 3 and '*ageres*' to 10 for this tutorial.

With all the parameters now set, we can use the '*fitjpmodel*' command to start the fitting process.

```
injectionindex
0.77
myears
50
bfield
0.6e-9
levels
3
ageres
10
fitjpmodel
0
```

The fitting process should take around 1-2 hours for the full data set on a standard desktop or laptop, far quicker (on the order of tens of minutes) for the small region or machines with a large number of cores.

Once completed, various statistical values (Figure 2.6) should be output to the terminal such as χ^2, maximum and minimum ages, their associated errors and binning statistics for the fit (see Section 3.4.3 and Harwood et al., 2013 for more details on binning statistics). We find that we recover a maximum age for the source of 40.97 megayears (full source) compared to the original (pre-convolved) 40 megayears of the simulation, well within the $\sim$2.5 megayear errors of this region.

15
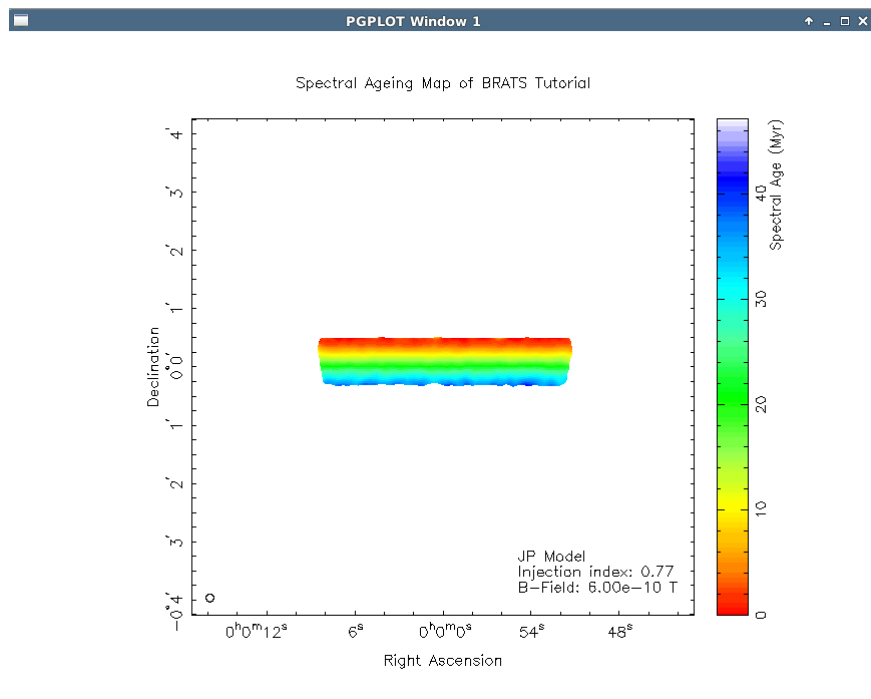
Figure 2.7: Spectral ageing map for the full tutorial source using the JP model of spectral ageing.

2.5 Mapping and plotting

In order to interpret the vast number of model fits that BRATS produces, a number of mapping and plotting tools are available. The complete list of plotting parameters can be found in Section 3.5.

The first plot one normally wants to create at after running a model fit is a spectral age map to visualise how the ages are distributed as a function of position. This can be done by using the '*specagemap*' command and simply choosing the data set and model fit which you wish to display.

```
specagemap
0
1
```

It is often useful to have the flux contours overlaid to these maps to better interpret them. Using the '*contours*' command and sticking with the default setting for now, we can rerun the spectral age map to see how they vary in relation to the source flux.

```
contours
specagemap
```

```
0
1
2
```

We see that there is a clear gradient in the spectral age running from low ages near the bright acceleration regions to the much fainter, oldest age regions which is what one would expect to observe for this simple simulated shock with downstream spectral ageing (Figure 2.7).

Once we have looked at the age distribution, the next logical step is to visualise the goodness-of-fit of the model and determine whether there are any specific regions of the source which are poorly fitted. There are two ways this can be done. The first is using the '*chisquaredmap*' command to plot the χ^2 values as a function of position. We turn off the contours again to give us a clearer view of the edge regions.

```
contours
chisquaredmap
0
1
```

We see that while the χ^2 values relative to each other increase as we move downstream the absolute values are low across the entire source and can be confident that our statistical values are representative of the source as a whole (Figure 2.8). This relative increase is due to the superposition of ages created when the maps were convolved to a larger beam size (or in real data, where the beam size is larger than regions of plasma of a single age) which has a greater impact on the old, highly curve spectra. Using small, pixel sized regions helps to reduce this effect by weighting the data towards the correct value but cannot be fully removed without the use of 'perfect' resolution images.

The second way of visualising this data is to plot the age errors as a function of position through the '*errormap*' command and selecting the data set, model and the type of errors we want (in this case the positive errors).

```
errormap
0
1
0
```

We see that the errors over the majority of the source are in the region of 1-2 megayears which is what one would expect for the given the number of images used and their associated errors (Figure 2.9).
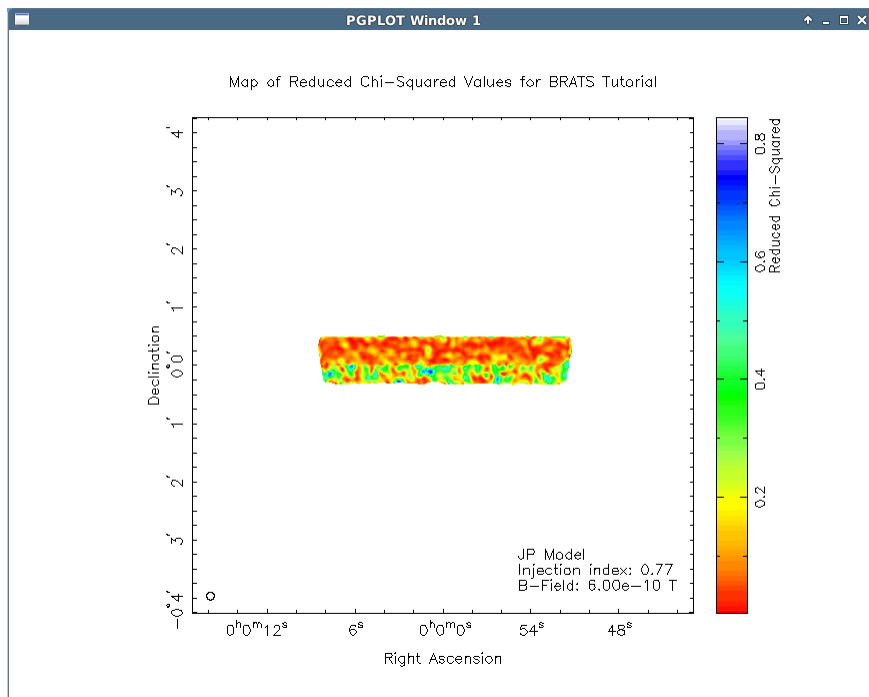
Figure 2.8: χ^2 map for the full tutorial source using the JP model of spectral ageing.

A few regions around the edge of the source have particularly high errors, but these are localised to a few small, low age regions with steep flux gradients and so would be unlikely to cause any major problems if we were to go on to do a full scientific analysis of the source.

It can also be useful (particularly when there are poorly fitted areas) to plot the models overlaid to the flux values as a function of frequency which can be achieved through the '*plotmodelobs*' command. However, due to the large number of regions we are dealing with ($\sim$10000 for the full source), it is normally advisable to only view a subsection of the data by setting the '*skip*' parameter. Here, we will set this to 1000 to produce 9 plots. If you are using the small region file you may wish to change this to 100 instead to produce a similar number.

```
skip
1000
plotmodelobs
0
1
```

This form of plot can be adjusted using the standard parameters detailed in Section 3.5.4. Now that
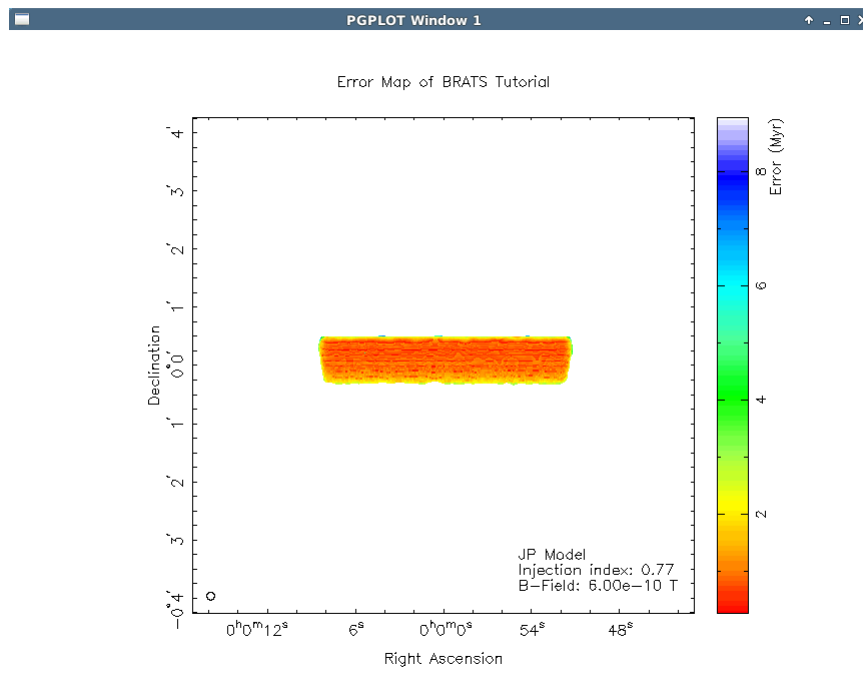
Figure 2.9: Error map for the full tutorial source using the JP model of spectral ageing.

we have seen how the data look and are confident that the fitting has been successful, we move on to the final part of the tutorial, backing up our model fits and exporting the data for further analysis.

2.6 Exporting

Once a model fit has run, it is advisable that we make a backup of that data, especially when it is a long run of a large source. This can be achieved through the '*fullexport*' command which outputs all of the source data, model fits etc. to a file in a custom .brats format to the data folder in your working directory. The volume of data exported can be controlled via the '*exportcompression*' parameter, but in nearly all cases the default settings are optimal.

```
fullexport
0
tutorialdatabackup
```

To import a saved data set, one simply uses the '*fullimport*' command to retrieve the information. Note you will need to modify the command below to the file name you have just exported if you wish to test this feature.

```
fullimport
./data/some_source_name_and_timestamp_tutorialdatabackup.brats
```

Although keeping a backup of the data set is an important step, having all of the information compacted into a single file is less than ideal for the purposes of analysis and presenting your results. BRATS therefore provides two ways of exporting data in a more sensible manner for this purpose. The exporting of visual data (e.g. plots and maps) is done through the '*export*' command which turns on and off the exporting of images which are normally output to screen to the images folder of your working directory. Here we will turn remake the spectral ageing map from the previous section, but this time with exporting turned on.

```
export
specagemap
0
1
```

Note that turning on exporting automatically activates an request for additional information (i.e. a unique file name) in some main commands. To turn off exporting and resume outputting images to screen, simply rerun the '*export*' command.

The exporting of numerical values is handled through the '*exportdata*' command. This is particularly useful for exporting information such as the values from the '*findinject*' to plot the χ^2 curve, or the spectral ages of a model fit which, when combined with exported region information, can be used to replot the spectral ageing map in your preferred style or colour scheme. Here, we will export the ages from the JP model fit we performed earlier.

```
exportdata
0
0
tutorialdataages
```

The results will be exported to the data folder in your working directory in text format.

Many other functions are also available in BRATS such as combining maps in the image plane, creating images of a source extrapolated from a given model and the fitting of integrated fluxes, as well as a wide variety of parameters to control how the data is fitted and presented (see Chapter 3), but we hope that this tutorial has provided the tools needed to get you started with BRATS. If you have any comments or spot any mistakes within this tutorial, please email brats@askanastronomer.co.uk or contact the author directly.

3

Detailed BRATS usage

3.1 Commands and conventions

Control of BRATS is achieved through a series of descriptive terminal commands. The following sections describe these commands and, where appropriate, the underlying principles and assumptions behind them. The tasks and parameters used to control BRATS can be divided into 3 categories; the primary commands, parameters set prior to running the command and parameters set within the main command. As of version 2.4.0, task run from the main command prompt are no longer case sensitive. Earlier versions require that all commands are in lower case. Note that the text escape command used within certain functions (*esc*) remains case sensitive in all versions.

Primary commands run the main functions of BRATS, for example, typing '*specindex*' into the main BRATS command prompt will create a spectral index map. The parameters of these primary commands are controlled by the second two categories. Those parameters which are either common to a range of primary commands (e.g. magnetic field strength) or are unlikely to change on a regular basis (e.g. gamma min) are contained within their own command which sets the value globally for all future tasks. For example, typing '*bfield*' into the main BRATS command prompt will ask the user to enter a new magnetic field strength. Once set, this new value will be used for all future model fitting runs. Parameters which are likely to change on a regular basis (e.g. the data set to be used) fall under the final category and are entered within the specific primary command. For example, upon running the primary command '*load*', the user will be promoted to enter the location of the data set which is local to only that single instance of the load command. It is also useful to note that certain reserved keywords can also be used within these requests for parameter values, e.g. '*ls*' can be used during the load command to list the current directory. Such values are noted accordingly in the following sections.

3.1.1   Basic commands

Command / input

brats -version .... : (When starting BRATS) Display the version information.

brats -help .... : (When starting BRATS) Display basics startup help and a link to the latest version.

quit .... : Quit BRATS.

help .... : Starts the BRATS help terminal which describes all of the available commands and parameters along with a brief summary of their usage.

esc .... : Escape to the command screen from (most) tasks requiring a text based input.

-1 .... : Escape to the command screen from (most) tasks requiring a numeric input.

888 .... : Select all data sets, images, models etc.

ls .... : List the contents of the current directory. Can also used where tasks require a text based input e.g. a directory location.

shell .... : Enter shell command prompt mode which allows the use of multi word system commands e.g. mkdir mynewimagesfolder

list .... : Output to the terminal all of the currently loaded datasets.

props .... : Output to terminal the basic properties of a dataset.

zoom .... : Set the zoom level to use when outputting maps etc.

shiftimage .... : Shift the centre of the output image by a give number of pixels in the X and Y direction e.g. for sources not at the pointing centre that require the zoom function to be used. NOTE: Currently not applied to the setregions command.

3.2 Loading data and image requirements

Command

load .... : Loads a new dataset into BRATS.

Parameters set prior to running command

sigma .... : Change the initial source detection level. Calculated using the RMS of the background region with no on-source multiplier applied (DEFAULT = 5).

casadata .... : Select which reduction type should be defaulted to if load cannot automatically determine the header type (DEFAULT = CASA).

telescope .... : Change the default flux calibration errors set when loading a map to a given telescope, or, a fixed user value (DEFAULT = JVLA).

Parameters set within the main command

directoryname .... : Location of directory containing images to be loaded.

backgroundregion .... : Location of the DS9 region file defining the area in which the off-source background noise should be calculated.

sourceregion (optional) .... : Location of the DS9 region file encompassing the target source (recommended).

redshift .... : Redshift of the target source. Required for spectral age fitting. For non-spectral age analysis where the redshift is not known, any reasonable dummy value can be entered.

To begin analysing a data set using BRATS, a series of FITS images of the source at varying frequencies must be loaded. The software acts independently of the method of reduction of the data but the images must contain compatible FITS headers [1]. The radio maps should be at a minimum matched in dimensions, coordinate system and beam size, along with being well aligned. These images should be

---

[1]Currently AIPS and CASA, contact the developer if you require an additional header type

located in a folder containing **only** the FITS files that are to be loaded and, ideally, be easily locatable (e.g. Figure 1.1). Any additional files in this directory will cause a failure of the data to load.

Note that due to a quirk in the FUNTOOLS library used to read the FITS files, there is an upper limit on the size of the image header. If exceeded, BRATS will exit with the error 'no WCS information in file while parsing filter at: XX:XX:XX.XXX'. This normally occurs due to a large history accumulated during reduction and can be fixed by running the AIPS Stalin command or by setting HISTORY = False when exporting images from CASA.

A region file is also required to identify where a value for background RMS noise should be calculated. This file must be in DS9[1] format and WCS coordinates. A second region file is also recommended, loosely encompassing the area in which the target of interest is located. This additional region file is vital where other bright sources are located within the image, but also reduces the computational time required in all images by ignoring the area outside of the defined region. This secondary region may also be used to exclude areas within the source which are not of interest, or that may interfere with later statistical analysis (e.g. AGN cores in analysis of radio galaxy lobes). Background and source region selections can be made using any software capable of saving region files in DS9 format such as DS9 itself or in CASA. Once the data and region files are in place a set of (optional) parameters may be adjusted before loading, such as the sigma cut-off level ('*sigma*').

The radio data are loaded into BRATS by specifying the location of the FITS directory and the region files described above, along with the redshift of the target source to the '*load*' task[2]. When run, the FITS header information is loaded using the FUNTOOLS library to access the image files and checks are undertaken to identify potential problems (e.g. identical map frequencies). The RMS noise of each map is then calculated and the flux values which fall within the defined region (if specified) and above the specified sigma cut-off level are loaded into memory. Note that the cut-off level must be set **before** a load is run using the 'sigma' command. A flux map of the first image in the array is then displayed to screen as a visual check that the expected data have loaded successfully. If required, this processes can be repeated to load multiple data sets which can be acted on either individually or as a whole by subsequent tasks. Unless otherwise stated, the functions described in this cookbook act only on the data stored in memory, leaving the original FITS images unchanged.

---

[1]http://ds9.si.edu/site/Home.html

[2]Note that 'ls' can be used at any point during the load process to list the contents of the current working directory

3.3 Adaptive regions

Command

setregions .... : Determines and sets adaptive regions for a given data set.

setsingleregion .... : Applies a single region to a selected data set.

Parameters set prior to running command

signaltonoise .... : Sets the signal-to-noise level that a region must reach. A value of 1 creates regions on a pixel-by-pixel basis (DEFAULT = 1).

sigma .... : Change the initial source detection level. Calculated using the RMS of the background region with no on-source multiplier applied. (DEFAULT = 5)

searcharea .... : Sets the maximum area a region can be before it is flagged (DEFAULT = 1).

fluxcalerror .... : Sets the flux calibration error for a single or range of maps and datasets. Automatically determined for a (limited) number or radio telescopes.

viewerrors .... : Displays the currently set values for the flux calibration errors.

onsource .... : Change the value by which the RMS is multiplied for the on-source noise (DEFAULT = 3).

hotpixels .... : Sets the limit at which pixels will be considered hot and cold as a decimal fraction of the surrounding pixels flux (DEFAULT = 0.2).

maptomap .... : Set the limit at which map-to-map variations will cause flagging as a decimal of the flux (DEFAULT = -1 [OFF]).

Parameters set within the main command

dataset .... : Select for which data set regions should be determined.

26

Once a data set has been loaded, regions which define the source and the area over which subsequent commands will be applied must be set (these should not be confused with the broad regions defined during loading). This task (or another region selection command) must be run before any fitting functions are performed on a data set.

The spectral analysis of radio sources, particularly in the context of spectral ageing, has traditionally required the use of large spatial regions in order to obtain the required signal to noise levels. With the increased performance of the new generation of radio interferometer and the consequent improvement in image fidelity, it is possible to consider much smaller regions than has previously been viable. In the case of bright sources with good *uv* coverage, it is often possible to consider the source spectra on a pixel-by-pixel basis; however, for old or poor quality data, one may need to consider larger regions to get the required signal to noise. The '*setregions*' function is therefore used to group pixels in to regions based on a specified set of parameters. These adaptive regions are constrained in two primary ways; signal to noise ratio and maximum search area.

For each pixel for which a region has not yet been defined, and has not been excluded by one of the bad pixel detection techniques (see below), its flux density S_{reg} is compared to a minimum flux density given by

$$S_{reg} \geq R_{SN} \left\{ (J \times S_{RMS}) \sqrt{n_{reg}/a_{beam}} \right\} \quad (3.1)$$

where R_{SN} is a user defined signal to noise ratio, J is the on-source noise multiplier (a multiplier for the off-source RMS value), S_{RMS} is the thermal noise, n_{reg} is the number of pixels currently in a given region and a_{beam} is the primary beam area. If the pixel flux density is below that of the minimum value, an adjacent pixel is added to the region and the test repeated with an increased value of n_{reg}. This process is then repeated adding the closest unused pixel until either the inequality of Equation 3.1 is satisfied, or n_{reg} is greater than a defined maximum search area. In the case that a pixel failed to satisfy these criteria it is marked as bad and is no longer considered in any subsequent analysis.

The adaptive regions function also provides additional bad pixel detection techniques above those of the signal-to-noise ratio and sigma level detections. A 'hot' and 'cold' pixel detection function can also be applied. When a pixel is tested for inclusion within a region its flux density value is compared with the surrounding pixels. If the tested flux density differs from those surrounding it by a specified multiple more then 50 per cent of the time, or, too many of the surrounding pixels were already marked as bad so the test cannot be performed reliably, the tested pixel is also excluded. It is important that the parameters for this detection method are set so as to only exclude large variations in flux density which are nonphysical and highly likely to be a result of spurious data. In a similar manner, variations across

maps can also be considered using setting the 'maptomap' command. This feature is only recommended where a smooth transition is expected and the seperation between images in frequency space is small. Note that such jumps are often a result of poor image alignment and are therefore often an indicator of a more fundemental, underlying problem with the data.

3.4 Model fitting and parameter determination

3.4.1 Spectral index

Command

specindex .... : Calculate and map the spectral indices by region of a given data set.

specchisquared .... : (Weighted GSL method only) Map the chi-squared values for the spectral index fitting as a function of position.

Parameters set prior to running command

specindexcalctype .... : Set the method used when fitting spectral indices to a data set (DEFAULT = 2 [weighted GSL least squares fit]).

printindex .... : Turn on and off the output of spectral index values to the terminal window as they are calculated (DEFAULT = OFF).

Parameters set within the main command

dataset .... : Select for which data set spectral indices should be determined.

The most straight forward form of analysis that can be performed is the fitting of spectral indices to the observed flux values. BRATS is able to use various methods to determine the spectral index for each

region, the most simple being a linear least squares regression in log space by the equation

$$\alpha_r = \frac{N \sum\limits_{i=0}^{N} (\log \nu_i \log S_{i,r}) - \sum\limits_{i=0}^{N} \log \nu_i \sum\limits_{i=0}^{N} \log S_{i,r}}{N \sum\limits_{i=0}^{N} (\log \nu_i)^2 - \left(\sum\limits_{i=0}^{N} \log \nu_i \right)^2} \quad (3.2)$$

where N is the total number of radio maps, i is the individual map index, r is the region number, ν is the map frequency and S is the region flux.

The results are then plotted as a function of position to form a spectral index map. As well as being a useful analysis tool on their own right, these maps provide a good initial check for the data quality and to identify any potential problems such as alignment issues without the need for use of the more computationally expensive spectral age model fitting.

As the uncertainty on the flux measurements of radio observations varies between images, observations and telescopes the constant errors assumed by a standard least squares fit is often not valid. BRATS therefore includes a weighted least squares option for spectral index determination. This uses the GSL 'wlinear' and 'linear_est' functions[1], where the weights are given by $w = 1/\sigma^2$ and σ is the error on a given flux measurement. Note that as fitting is performed in log-log space, **the fractional errors on the measurements must be small** so as that one can assume that the errors remain approximately sysmetric in natural log space. If neither of the assumptions made by these fitting types can be assumed and you would like an additional method included, please contact the developer.

At the time of writing, weighted least squares is the default fitting type for spectral indices. The desired method can be selected using the '*specindexcalctype*' command.

3.4.2   Spectral age fitting

Command

fitjpmodel .... : Fit the JP model of spectral ageing to a given data set.

fitkpmodel .... : Fit the KP model of spectral ageing to a given data set.

fitjptribble .... : Fit the Tribble model of spectral ageing to a given data set.

fitintegrated .... : Fit a model to integrated flux values. Note integrated fitting values are

---

[1]https://www.gnu.org/software/gsl/manual/html_node/Linear-regression.html#Linear-regression

output in text format only (see end of this section for details).

Parameters set prior to running command

bfield .... : Set the magnetic field strength to use in model fitting (DEFAULT = 1e-9 T).

injectionindex .... : Set the injection index to use in model fitting (DEFAULT = 0.6).

modelpower .... : Alternative to '*injectionindex*'. Sets the injection index to use in model fitting based on the model power, where $\alpha_{inj} = (P - 1)/2$ (DEFAULT = 2.2).

myears .... : Set the maximum spectral age in megayears which for which fitting should be attempted (DEFAULT = 10).

ageres .... : Change the resolution of the ages attempted when fitting models e.g. 10 gives a step size 0.1 for the first level of fitting between 1 and 10 myears (DEFAULT = 10).

levels .... : The number of (age) levels deep to go when model fitting. 3 is recommended for testing and initial results, 5 for final results. Note setting this to high values drastically increases computation time! (DEFAULT = 3).

gmin .... : Set the minimum value of gamma for model fitting (DEFAULT = 10).

gmax .... : Set the maximum value of gamma for model fitting (DEFAULT = 1000000).

printresults .... : Turn on and off the printing of individual results when model fitting. The final summary will still be printed on completion of a model fit (DEFAULT = OFF).

printreject .... : Turn on and off the output whether it should be stated that a model should be rejected if the average chi-squared is above the 90 per cent confidence interval. This only affects the print out to the terminal in the summary, not the fitting itself. Note if there are a number of regions where you expect a bad fit e.g. from dynamic range issues, the average may not be a suitable value to use (DEFAULT = OFF).

Parameters set within the main command

dataset .... : Select for which data set the spectral age model should be fitted.

model .... : (fitintegrated only) Select which model to be fitted.

redshift .... : (fitintegrated only) Enter the redshift of the source. Note that for standard non-integrated fitting, redshift is set during the '*load*' command.

numberoffrequencies .... : (fitintegrated only) Number of data points to be fitted to.

frequency .... : (fitintegrated only) Enter the frequency of the nth data point.

flux .... : (fitintegrated only) Enter the flux of the nth data point.

error .... : (fitintegrated only) Enter the error of the nth data point.

Currently, 3 models of spectral ageing can be fitted by BRATS; the JP, KP and Tribble models ('*fitjpmodel*', '*fitkpmodel*' and '*fitjptribble*' respectively). In contrast to classical methods of determining spectral age such as the colour-colour plots, BRATS is capable of carrying out the full numerical integrations required to determine the flux values at a given frequency and age. For a user defined magnetic field strength ('*bfield*') and injection index ('*injectindex*'), these model fluxes are then compared to the observations and a χ^2 value calculated to determine the goodness-of-fit (see Section 3.4.3). This process is then repeated over a range of ages and a best fitting spectral age determined for each region within the source.

To determining the model fluxes, the standard synchrotron equations are used (e.g. Longair, 2011), where the emissivity of a single electron as a function of frequency is given by

$$J(\nu) = \frac{\sqrt{3}Be^3\sin\alpha}{8\pi\epsilon_0 cm_e}F(x) \tag{3.3}$$

To reduce computational time, a look-up table of the single-electron synchrotron radiation spectrum values, $F(x)$, is made between $x = 1 \times 10^{-4}$ and 22 at 100 logarithmic intervals. The function $F(x)$ is taken as defined by Rybicki & Lightman (1979) to be

$$F(x) = x\int_x^\infty K_{5/3}(z)dz \tag{3.4}$$

where K is the Bessel function of order 5/3, $x \equiv \nu/\nu_c$ and $\nu_c = \gamma^2 eB_\perp/2\pi m_e$ is the critical frequency. These points are then used to find any value of x using a log-linear two-point interpolation. For cases where x falls outside of the minimum tabulated range, the asymptote of Pacholczyk (1970)

$$F(x) = 2.15x^{1/3} \tag{3.5}$$

is used and for large values of x it is assumed that $F(x) = 0$. The pitch angles are taken to be isotropic, hence integrating over the probability distribution $(1/2)\sin\alpha$ and between the minimum (E_{min}) and maximum (E_{max}) electron energies the total model emission is found by

$$S_{model}(\nu) = \frac{\sqrt{3}e^3 B}{8\pi\epsilon_0 cm_e}\int_0^\pi\int_{E_{min}}^{E_{max}} F(x)\frac{1}{2}\sin^2\alpha\, n_e(E)\, de\, d\alpha \tag{3.6}$$

31

where n_e is the electron energy distribution subject to losses given by

$$n_e(E) = N_0 E^{-2\alpha+1}(1 - \beta)^{(2\alpha+1)-2} \tag{3.7}$$

and β are the model dependent losses. For the KP model

$$\beta_{KP} = B^2 Et \left(\frac{4\sigma_T \sin\alpha}{6m_e^2 \nu_c^3 \mu_0} \right) \tag{3.8}$$

and for the JP model

$$\beta_{JP} = B^2 Et \left(\frac{4\sigma_T}{6m_e^2 \nu_c^3 \mu_0} \right) \tag{3.9}$$

In the case of the free-streaming Tribble model (Hardcastle, 2013; Harwood et al., 2013; Tribble, 1993), the JP losses are integrated over a Maxwell-Boltzmann distribution, hence where B_0 is the mean magnetic field strength of the source Equation 3.6 becomes

$$S_{model}(\nu) = \sqrt{\frac{2}{\pi}} \frac{\sqrt{3}e^3}{8\pi\epsilon_0 cm_e} \int_0^\infty \int_0^\pi \int_{E_{min}}^{E_{max}} F(x)\sin^2\alpha\, n_e(E)$$
$$\times \frac{B^3 \exp\left(-B^2/2B_0^2\right)}{B_0^3}\, dE\, d\alpha\, dB \tag{3.10}$$

Using the GNU Scientific Library[1] (GSL) for the required numerical integration, unnormalized flux density values are determined between a user defined minimum and maximum age range controlled via the '*myears*' command. These values are then normalized (see below) and a χ^2 statistical test performed using the standard equation

$$\chi^2 = \sum_{\nu=1}^N \left(\frac{S_{i,\nu} - S_{model,\nu}}{\Delta S_{i,\nu}} \right)^2 \tag{3.11}$$

where N is the total number of observed frequencies, $S_{i,\nu}$ is the observed flux density of region i at frequency ν, $S_{model,\nu}$ is the model flux and $\Delta S_{i,\nu}$ is the total uncertainty of the observed region given by

$$\Delta S_{i,\nu} = \sqrt{\left\{ (J \times S_{RMS})\sqrt{n_{reg}/a_{beam}} \right\}^2 + (S_{i,\nu} \times \Delta S_E)^2} \tag{3.12}$$

where ΔS_E is the flux calibration error for a given observation and J is the on-source noise multiplier.

As one cannot be certain that in all cases the spectral age will not contain local minima, a grid search is used to find the best fitting spectral age for each model. In order to obtain a high accuracy, but at a reasonable computing cost, a broad search is first run using large age intervals. This process is then automatically repeated over a series of decreasing age intervals bounded by the best fitting model of

---

[1]http://www.gnu.org/software/gsl/

the previous cycle until the required accuracy is obtained. These search parameters are controlled by using the '*ageres*' and '*levels*' commands respectively. For each spectral age tested the normalization is determined by a golden ratio search based on the methods of Press et al. (2007). Using this approach over a 2-dimensional grid search vastly improves the speed of the model fitting and allowed accuracy to be achieved to the floating point limit.

As the method described above provides the χ^2 curve of each region as a function of age, the uncertainty in the spectral age can also be calculated. Avni (1976) shows that a 1σ error is given by a deviation of $\Delta\chi^2 = 1$ from the minimum χ^2 value. These errors are therefore calculated during the fitting process and can then be mapped and exported for further analysis using the '*errormap*' and '*exportdata*' commands respectively (also see Sections 3.5.1 and Sections 3.6).

Once fitting has complete, a range of statistical values (see Section 3.4.3) are output to screen and spectral age mapping becomes available. Using the '*specagemap*' and '*chisquaredmap*' functions, the spectral age and goodness-of-fit values can be plotted as a function of position (also see Section 3.5.1). An example spectral age map (from simulated data) is shown in Figure 3.1.

It is also possible to fit models to the integrated flux values of a source using the '*fitintegrated*' command. This function does not require map to be loaded but instead uses user entered value when the command is run. The user should, however, carefully consider the validity of any model fitted to the integrated flux of a source e.g. problems due to the superposition of populations of an ageing plasma (see Harwood et al., 2013; Stroe et al., 2014 for examples and detailed discussion of some of these problems). As this form of fitting single injection models to integrated flux values has limited applications, results are currently only available in text format automatically output to the data folder of the users working directory. If (or more likely when) additional models are added which require the use of the integrated flux, such as continuous injection models, this feature will be expanded to include the full set of plotting and statistical features. The timescale on which this will occur is likely to be determined by demand, so if you wish to see such models added please contact brats@askanastronomer.co.uk and let us know.

3.4.3   Statistics

BRATS provides a number of statistics to aid in the interpretation of the model fitting described above. The primary measure for an individual fit of a region is given by its χ^2 value (Equation 3.11). Although the plotting of these χ^2 values as a function of position is an important tool in determining the goodness-of-fit for regions within a source and identifying area that are poorly described by a model, additional statistical tests are required to infer the goodness-of-fit of a given model over the source as a whole.
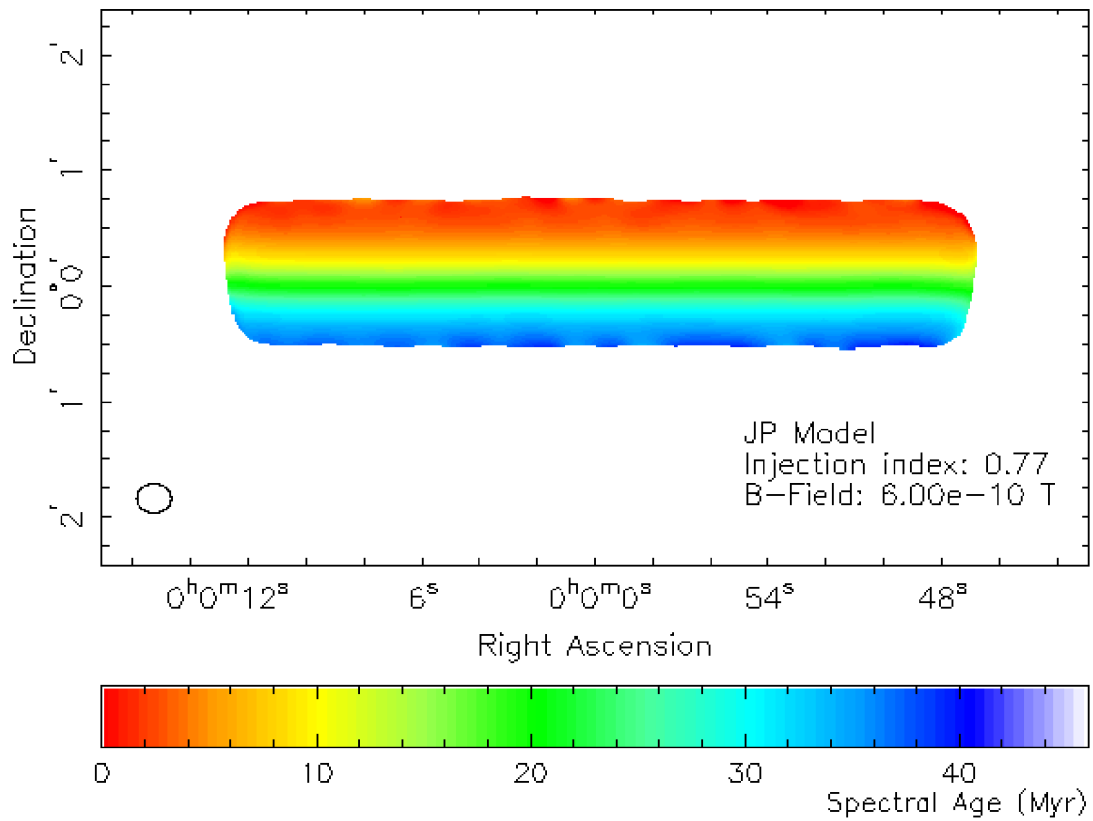
Figure 3.1: Fitting of the JP model of spectral ageing to simulated observations. The simulated data consist of seven images between 153 MHz and 2.27 GHz determined from an ideal JP model with $\alpha_{inj} = 0.77$. Flux values are determined on a pixel by pixel (1 arcsec) basis between 0 and 40 Myrs, then convolved with a 15 arcsec beam. The resulting images were then analyzed with the BRATS '*fitjpmodel*' command to produce the above spectral ageing map, which recovers the input ages well.

The sum of the χ^2 for an entire source can provide an initial indication of the overall goodness-of-fit; however, regions where there are high χ^2 values for known, non-physical reasons (such as dynamic range effects) can cause the average to be biased toward the rejection of a model. Two additional statistical checks are therefore available to account for this bias. The χ^2 cumulative distribution function is first calculated at 68, 90, 95 and 99 per cent confidence levels using the computational techniques of Press et al. (2007) and the GNU Scientific Library functions. The χ^2 value of each region is then binned to a given confidence level with a cut-off for the rejection of the region set at ≥ 95 per cent confidence. If more than half of the regions fall above this rejection cut-off, the model over the source is classed as a poor fit and rejected. The median of the individual rejected or non-rejected interval bins is then taken to give a level at which the (non-) rejection is made. For example, if the median of the non-rejected regions falls within the 68 to 90 per cent interval bin, we can say that the model for the source as a whole cannot be rejected at greater than a 90 per cent confidence level. This method of analysis allows the goodness-of-fit to be weighted towards the area over which a model is well or poorly fitted rather than a bias towards high χ^2, but potentially small, regions of poor fit.

3.4.4 Image reconstruction (model extrapolation) and difference maps

Command
mapfrommodel .... : Creates a simulated radio map in FITS format at any user defined frequency given any spectral ageing model. Model fitting of a real data set must first be performed to obtain the required spectral ages and normalisations.

Parameters set prior to running command
None

Parameters set within the main command
dataset .... : Select which data set should provide the required spectral ageing model fitting data.

frequency .... : Set to what frequency the spectral ageing model should be extrapo-
lated / iterpolated.
name .... : Set a name from the resulting reconstructed image file.

Command
diffmap .... : Subtracts the flux of one FITS file from another and outputs a third
FITS image. Subtraction can either be full map or confined to the regions of a data set.

Parameters set prior to running command
None

Parameters set within the main command
sourcemap .... : Image which should be subtracted from.
subtractionmap .... : Image to subtract.
restricttoregions .... : Sets whether to restrict the subtraction to the regions of a given data
set.

Two useful features related to model fitting are the image reconstruction ('*mapfrommodel*') using extrapolated or interpolated spectral ageing models and the difference map function ('*diffmap*') functions. For a radio source well described by a spectral ageing model of known age and normalization, it is theoretically possible to produce an image at any given frequency. Once a spectral ageing model has been fitted using the methods of Section 3.4.2, the BRATS image reconstruction function can be used to perform such a task. Using the best fitting spectral ages and normalization, BRATS determines the flux values of a given model for any user defined frequency. This is particularly useful for the prediction of source behaviour at previously unobserved frequencies and for comparison to combined or alternative images to determine the model accuracy.

BRATS facilitates such comparisons by providing the difference map function. This task loads two FITS images (these do not have to have been produced or fitted by BRATS) through the standard method detailed in Section 3.2. The fluxes of the first map are then subtracted from the second and a FITS image with the residual flux values is output. This is particularly useful in determining how well a 'pure' spectral ageing model describes a source.

3.4.5 Injection index determination

Command

findinject .... : Run a model fit (or multiple model fits) with varying injection indices in an attempt to find the best fitting injection index. Warning: This function can take a VERY long time to run, especially if multiple models over multiple datasets are being attempted.

Parameters set prior to running command

mininject .... : Set the minimum injection index be tested with findinject (DEFAULT = 0.5).

maxinject .... : Set the maximum injection index be tested with findinject (DEFAULT = 1.0).

injectintervals .... : Set the number of intervals between mininject and maxinject to be tested by findinject. n + 1 intevals are tested to find the best fit e.g. with mininject 0.5 and maxinject 1.0, n = 10 will result in steps of 0.5 (DEFAULT = 10).

bfield .... : Set the magnetic field strength to use in model fitting (DEFAULT = 1e-9 T).

modelpower .... : Alternative to '*injectionindex*'. Sets the injection index to use in model fitting based on the model power, where $\alpha_{inj} = (P - 1)/2$ (DEFAULT = 2.2).

myears .... : Set the maximum spectral age in megayears which for which fitting should be attempted (DEFAULT = 10).

ageres .... : Change the resolution of the ages attempted when fitting models e.g. 10 gives a step size 0.1 for the first level of fitting between 1 and 10 myears (DEFAULT = 10).

levels .... : The number of (age) levels deep to go when model fitting. 3 is rec-
ommended for testing and initial results, 5 for final results. Note setting this to high values drastically
increases computation time! (DEFAULT = 3).
gmin .... : Set the minimum value of gamma for model fitting (DEFAULT = 10).
gmax .... : Set the maximum value of gamma for model fitting (DEFAULT =
100000).

Parameters set within the main command
dataset .... : Select for which data set the injection index should be determined.
ageingmodel .... : Select which spectral ageing model should be fitted.

BRATS also provides the ability to determine the injection index of a given source through the '*find-inject*' command. This function fits models using the methods in the previous section, but in addition also varies the injection index over a range defined by the '*injectmin*', '*injectmax*' and '*injectintervals*' commands. On completion, the sum of the χ^2 over all regions for each injection index can be output via the '*exportdata*' command, allowing a goodness-of-fit curve to be determined as a function of injection index. The minimum of this curve in turn gives the best fitting injection index for a given model and source. As this method of determining the injection index can use all of the observed flux, it is not reliant on measurements made of the relatively small particle acceleration regions, or the assumed location at which this particle acceleration occurs.

The errors for the injection index can be calculated in a similar manner to those described for the spectral ages in Section 3.4.2. However, as here one uses the sum of χ^2 over all regions for each injection index, the values are over weighted by a factor of the beam area. Assuming the injection index is approximately constant across any given source, the injection index errors are therefore determined by finding where $\Delta\chi^2 = 1$ for the corrected χ^2 values (χ^2_{cor}) such that

$$\chi^2_{cor} = \frac{\chi^2}{A_{beam}} \tag{3.13}$$

where A_{beam} is the beam area. Examples of the use of this method and discussion of their reliability (including comparison to simulations) are further given by Harwood et al. (2013) and Stroe et al. (2014).

38

3.5   Plots, maps and data visualisation

The large number of regions over BRATS performs its analysis mean that individual scatter plots, whilst useful in specific cases, are often a less than ideal way of representing the large volume of information gathered during model fitting and other spectral analysis functions. BRATS therefore provides a variety of visualisation options to help better understand and communicate these results. This Section describes these features and the parameters used to control them.

3.5.1   Spectral ageing model visualisation

Command

specagemap .... : Plot a map of the spectral age as a function of position.

chisquaredmap .... : Plot a map of the chi-squared values as a function of position.

errormap .... : Plot a map of spectral age error as a function of position.

Parameters set prior to running command

alwayszeroage .... : (specagemap only) Turn on and off fixing the minimum age range to zero for spectral ageing maps. Note: this only adjusts the wedge and colour range, not the data values. (DEFAULT = ON).

contours .... : Turn on and off the overlay of flux contours (also see Section 3.5.5).

Parameters set within the main command

dataset .... : Select for which data set model fitting results should be plotted.

model .... : Select for which model results should be plotted.

errortype .... : (errormap only) Select whether positive or negative errors should be plotted (positive errors recommended).

The visualisation of spectral age model fitting is an important tool in the analysis of a data set. The commands shown above provide the ability to map the age, chi-squared and errors as a function of position. This can either be to screen (default) or to an image file (see Section 3.6.1 for further details). If an alternative colour scheme or format is desired it is also possible to export the derived values for plotting via your preferred method, details if which are given in Section 3.6.2.

3.5.2 Standard plotting

Command

plotmodelobs .... : Plot the flux values of both the model and observed data as a function of frequency.

plotflux .... : Plot region flux and a function of frequency.

plotspecindex .... : Plot the spectral index fits for each region.

plotcurvespec .... : Plot curvature (fitpoly) against spectral index for a given dataset.

Parameters set prior to running command

Standard global plotting parameters (see Section 3.5.4)

Parameters set within the main command

dataset .... : Select for which data set the results should be plotted.

model .... : (plotmodelobs only) Select for which model results should be plotted.

Standard scatter plots can also be a key tool in the analysis of a data set. A range of useful plots are therefore provided within BRATS. These plots can either be to output to screen (default) or to an image file (see Section 3.6.1 for further details). Where a large number of plots exists (e.g. model values vs observed flux in a large 20,000 regions data set) it is recommended that the '*skip*' parameter is used to select a subset of the data (see Section 3.5.4).

3.5.3   Flux maps

Command

fluxmap .... : Create a standard flux map by plotting the flux of a source as a function of position. Note only flux above the original flux cut made when loading the data will be plotted.

Parameters set prior to running command

Standard global plotting parameters (see Section 3.5.4)

Parameters set within the main command

dataset .... : Select for which data set the flux should be plotted.

frequency .... : Select which frequency should be plotted.

rawvsregions .... : Select if the raw data or adaptive regions should be used when creating the flux maps.

average .... : (adaptive regions only) Select if the total or average flux of each regions should be mapped.

While many software packages are available for displaying the flux of a source, it is often useful to have this information quickly to hand when analysing results or tracking down problems. BRATS therefore provides the '*fluxmap*' command to facilitate this need. These maps can either be to output to screen (default) or to an image file (see Section 3.6.1 for further details).

3.5.4   Global plot parameters

Parameters set prior to running command

border .... : Set the border size in pixels for flux, spectral index and curvature maps

(DEFAULT = 4).

titles .... : Turn on and off titles in maps and plots.

labels .... : Turn on and off axis labels in maps and plots.

axismarks .... : Turn on and off axis markings in maps and plots.

autoscale .... : Turn on and off autoscaling of all plot axis, and set or change user defined values.

uselogs .... : Turn on and off the use of logarithmic values for plots.

skip .... : Set a value of how many regions to skip when plotting (i.e. only plot every nth region).

symbol .... : Change the symbol which is used to mark values when creating plots.

zoom .... : Set the zoom level to use when outputting maps etc.

shiftimage .... : Shift the centre of the output image by a give number of pixels in the X and Y direction e.g. for sources not at the pointing centre that require the zoom function to be used. NOTE: Currently not applied to the setregions command.

smoothmaps .... : Switch between smooth maps and the original style with harder cuts (DEFAULT = ON).

wedgemultiplier .... : Multiplier to apply to the maximim value for a given dataset that defines the upper limit of the wedge colour. This is useful in avoiding real values from taking the background colour (DEFAULT = 1.15).

scaletype .... : Set the scale to be used when mapping. Options are: Pixels, Arcseconds (from centre of map), Arcseconds in degrees, minutes, seconds format or WCS. (DEFAULT = WCS).

posmap .... : Set which map should be used as reference for co-ordinates when plotting maps in WCS (DEFAULT = 0).

beam .... : Turn on and off the plotting of the beam to maps (DEFAULT = ON).

beampos .... : Set the position to plot the beam when making maps (DEFAULT X = 10, Y = 10).

usereduced .... : Turn on and off the use of reduced chi-squared values for plotting and mapping (DEFAULT = ON).

The parameters described above are used to control the global properties of plotting within BRATS. These apply to both the mapping (e.g. spectral age maps) and standard plotting (e.g. model vs observed flux) commands.

3.5.5 Contours

Command

contours .... : Turn on and off the printing of contours (DEFAULT = OFF).

Parameters set prior to running command

contourlevels .... : Set the number of contours to plot between min and max values (DEFAULT = 8).

contourcolour .... : Set the colour of the contour lines (DEFAULT = BLACK).

contourlinestyle .... : Set the style of contour lines (DEFAULT = SOLID).

contourlogs .... : Set whether to use linear or logarithmic spacings between contours (DEFAULT = LOG).

firstcontour .... : The level of the first contour to plot (above the minimum) (DEFAULT = 1).

Parameters set within the main command

frequency .... : Set what frequency map should be used to determine the contours to be overlaid. An additional option will automatically appear for this selection to be made in each main command where contours are relevant e.g. the '*specagemap*' command.

Most images made in BRATS can be overlaid with flux contours by using '*contours*' command. When turn on, an additional option will automatically appear when running standard mapping commands such as '*specagemap*' and '*errormap*' commands. Contours are determined from the original flux data.

3.6 Exporting data and images

Many of the functions in BRATS, particularly model fitting of large data sets, can be time intensive. The ability to export data sets and the images they produce for publication purposes is therefore important feature. This section describes the commands which can be used to perform such tasks and some considerations which should be made when doing so.

3.6.1 Exporting images

Command

export .... : Turn on and off the exporting of data (DEFAULT = OFF).

Parameters set prior to running command

imageloc .... : Set the location for images to be exported (DEFAULT = ./images).

imagetype .... : Set the format of exported images (DEFAULT = PNG).

Parameters set within the main command

None

The '*export*' command can be used to control if the images and maps being produced by various command will be output to either to screen (off) or to a file (on). When set to on, parameters set via the '*imageloc*' and '*imagetype*' commands control the location (relative to the path in which BRATS was run) and type of file export respectively. Note that for many setups, the often favoured EPS format does not output correctly and may cause BRATS to exit unexpectedly. This problem seems to be system dependent and related to PGPLOT, something which we cannot correct directly. We are currently looking for a suitable solution for this problem but if encountered on your setup, we suggest exporting in PNG format and converting to EPS via your preferred imaging software e.g. GNU image manipulation tools.

3.6.2 Exporting data

Command

exportdata .... : Exports numerical data such as spectral ages, errors and injection indices to a text file.

Parameters set prior to running command

dataloc .... : Set the location for data to be exported (DEFAULT = ./data).

Parameters set within the main command

dataset .... : Select which data set for which the data should be exported.

datatobeexported .... : Select which data should be exported.

filename .... : Enter a name to be appended to the end of the standard naming format.

The '*exportdata*' command can be used to export various data values (spectral ages, chi-squared values, errors etc.) to a text file at a location set through the '*dataloc*' parameter. This is particularly useful for exporting injection indices determined through the '*findinject*' command for plotting. This can also be used to export values such as spectral ages which, when combined with the region location data, can be used to recreate images in alternative colour schemes and formats via your preferred plotting method. Note for full backups of data sets, use of the '*fullexport*' is recommended (see Section 3.6.3).

3.6.3 Exporting and importing full data sets

Command

fullexport .... : Export a full dataset, including any model fitting, in .brats format.

fullimport .... : Import a full dataset from a .brats format file.

Parameters set prior to running command

exportcompression .... : (Export only) Set exporting of a full dataset to either full or compressed. Note that compressed is recommended in nearly all circumstances. (DEFAULT = Compressed).

dataloc .... : (Export only) Set the location for data to be exported (DEFAULT = ./data).

Parameters set within the main command

dataset .... : (Export only) Select which data set should be exported.

importfile .... : (Import only) Select the .brats file from which the data should be imported.

The '*fullexport*' command can be used to export a full data set for backup or later use, which can then be reloaded into BRATS via the '*fullimport*' command. The exported file uses a custom .brats format and contains all information relating to the image values (flux, region data etc.) as well as 'long duration' fitting runs (spectral index, chi-squared values etc.). For disk space considerations, functions which are quick to run such as spectral index fitting, are not included (these can be exported via the '*exportdata*' command if required). The export function uses 2 format types, full and compressed, which are controlled via the '*exportcompression*' parameter. It is nearly always recommended that the compressed format is used, especially for datasets which use large or many FITS images. They key difference between the two formats is in the regions selection. In full format mode, the reason for each pixels inclusion or exclusion is included in the exported file often leading to very large file sizes. The compressed mode assumes that any non-included pixel is due to being below the flux limit, greatly reducing the size of the exported file. The type of file being imported (either full or compressed) is automatically determined by the '*fullimport*' command.

3.7   Other functions and parameters

3.7.1   Image combination

Command

combineimages .... : Combines a set of radio maps to a common (user defined) frequency in the image plane.

Parameters set prior to running command

None

Parameters set within the main command

imagefolder .... : Select a folder containing the images to be combined.
frequency .... : Set the frequency at which the combined map should be made.
powerlawscaling .... : Set the power law that the images should be scaled by.

A useful tool available in BRATS is the ability to easily combine radio maps in the image plane. The '*combineimages*' command first loads images into memory using the same methods as detailed in Section 3.2. The flux values are then scaled to a given frequency by a user defined spectral index and the values combined on a pixel by pixel basis. The fluxes are then averaged and the FUNTOOLS library is then used to export a new combined image. It should be noted that the proper alignment of the radio maps is an important factor in ensuring good quality in the resulting image.

3.7.2   Fixed regions

Command

fixedregions .... : Applies the regions of one data set to another using pixel coordinates.

Parameters set prior to running command

None

Parameters set within the main command

regionsfrom .... : Select which data set's regions should be used as reference.
regionsto .... : Select which data set the regions should be applied to.

The '*fixedregions*' command applies the regions which have been set for one data set to another. The mapping is done on a pixel by pixel basis and so it should be ensured by the user that all maps across both data sets have been properly aligned with each other. This is particularly useful when comparing the results of a model fit to that of a subsection of the original data set, where the excluded images are, for example, thought to be biasing the results.

3.7.3 Colour-colour plots

Command

colourcolour .... : Create a colour-colour plot of a given data set.

Parameters set prior to running command

None

Parameters set within the main command

dataset .... : Select which data set a colour-colour plot should be produced for.

lowestfrequency .... : Select a map to be used as the lowest frequency data point.

midfrequency .... : Select a map to be used as the mid frequency data point.

highfrequency .... : Select a map to be used as the high frequency data point.

A further method of analysis also available in BRATS, is the creation of colour-colour plots. This technique uses two, two-point power laws to describe the spectral curvature of a source over a large frequency range. This method of determining spectral age has met with some success (e.g. Hardcastle & Looney, 2001; Katz-Stone et al., 1993); however, as it is not the main focus of the software package and is mainly included for legacy purposes to allow comparison to previous studies, a full detailed explanation of the pros and cons of this technique is not given here.

3.7.4 Fitting polynomials

Command

fitpoly .... : Fit an nth order polynomial to the regions of a given data set.

Parameters set prior to running command

setpoly .... : Define the order of polynomial to use for fitting (DEFAULT = 2).

curvecon .... : Switch the sign convention used when fitting curves (DEFAULT = OFF).

plotres .... : Set the number of points to use when plotting the curve (DEFAULT = 1000, MINIMUM 10).

Parameters set within the main command

dataset .... : Select which data set the polynomial fit should be performed on.

Uses polynomial regression to fit the curves to a data set on a region by region basis. Results are output to a single plot at the end of the task. For large number of regions, set the 'skip' parameter before running. This function currently has basic functionality and is primarily used as a sanity check for data. If you are interested in using this function for a more detailed analysis, please contact the author either by email or via the website to discuss your requirements.

3.7.5 Example spectral ageing model plots

Command

plotjpmodel .... : Plot an example JP model between 'minmodelfreq' and 'maxmodelfreq' for an arbitrary normalisation.

plotkpmodel .... : Plot an example KP model between 'minmodelfreq' and 'maxmodelfreq' for an arbitrary normalisation.

plotjptribble .... : Plot an example Tribble model between 'minmodelfreq' and 'maxmodelfreq' for an arbitrary normalisation.

Parameters set prior to running command

minmodelfreq .... : Change the minimum frequency in Hz for example spectral ageing models (DEFAULT = 10^7Hz).

maxmodelfreq .... : Change the minimum frequency in Hz for example spectral ageing models (DEFAULT = 10^{13}Hz).

modelmyears .... : Set the maximum age to plot when outputting models (e.g. plotjpmodel). Models will output from 0 megayears to this number at intervals of 1 megayear.

Parameters set within the main command

None

BRATS also provides the ability to output (either to screen or an image file) example plots for each of the spectral ageing models for a given set of user parameters at an arbitrary normalisation. These functions use the standard model fitting parameters (e.g. bfield) with the exception of those listed above. This can be particularly useful for both presentation style images to display the general principles of spectral ageing, or as a visual test of how various model features change (e.g. the break frequency) with varying model parameters.

3.7.6 Resizing images

Command

resizeimage .... : Resize a series of FITS images to a new, user defined size.

Parameters set prior to running command

imageloc .... : Set the location for the resized images to be exported (DEFAULT = ./images).

Parameters set within the main command

imagefolder .... : Select which folder contains the images to be resized.

X size .... : Size of the new image's X axis (in pixels).

Y size .... : Size of the new image's Y axis (in pixels).

The majority of the BRATS commands require that a data set has been loaded containing images which are match in terms of their basic parameters e.g. image size. This can sometimes be problematic when dealing with data taken from various sources, especially when the original UV data is not available. Standard reductions tools such as AIPS and CASA can be used to match parameters such as resolution through smoothing, but systematically resizing a large number of images can often require a greater effort. The '*resizeimage*' command therefore provides the ability to resize images (in terms of pixels)

quickly and easily. These images can be cropped and expanded in either direction to a user specified sized at run time. In cases where the images are to be expanded in either one (or both) axis, pixels outside of the original data will be given a value of 0 Jy.

3.7.7 Scale flux

Command

scaleflux .... : Scales the raw flux values by a given scaling factor. Either single scaling value or interpolation / extrapolation of two values can be used for either all or a subsection of maps. Region selection should be re-run after the scaling has been applied.

Parameters set prior to running command
None

Parameters set within the main command

dataset .... : Select for which data set the flux should be scaled.

mapstobescaled .... : Select which maps within the data set should be scaled.

minfrequency .... : (Frequency range only) Select the minimum frequency range to be scaled.

maxfrequency .... : (Frequency range only) Select the maximum frequency range to be scaled.

maptobescaled .... : (Single map only) Enter the map to apply the scaling factor.

scaletype .... : Select the type of scaling to be used.

firstfrequency .... : (Interpolation / extrapolation only) Enter the frequency of the first scaling factor to be used.

firstscalingvalue .... : (Interpolation / extrapolation only) Enter the value of first scaling factor to be used.

secondfrequency .... : (Interpolation / extrapolation only) Enter the frequency of the second scaling factor to be used.

secondscalingvalue .... : (Interpolation / extrapolation only) Enter the value of second scaling factor to be used.

singlescalingvalue .... : (Single value only) Enter the value of the scaling factor to be used.

It is often necessary to scale the flux of a map due to, for example, a change in flux scale in archival data or to check the impact of a recent flux calibration change post data reduction. BRATS provides the ability to scale the flux values of either the entire data set, a specific map or over a defined frequency range via the '*scaleflux*' command. The scaling can be either a single fractional value or linearly interpolated / extrapolated between values at two given frequencies.

References

Avni Y., 1976, ApJ, 210, 642 33

Hardcastle M. J., 2013, MNRAS, 433, 3364 32

Hardcastle M. J., Looney L. W., 2001, MNRAS, 320, 355 49

Harwood J. J., Hardcastle M. J., Croston J. H., Goodger J. L., 2013, MNRAS, 435, 3353 9, 12, 15, 32, 33, 38

Katz-Stone D. M., Rudnick L., Anderson M. C., 1993, ApJ, 407, 549 49

Longair M. S., 2011, High Energy Astrophysics. Cambridge University Press 31

Pacholczyk A. G., 1970, Radio astrophysics. Nonthermal processes in galactic and extragalactic sources. San Francisco, Freeman 31

Press W. H., Teukolsky, S. A.and Vetterling W. T., Flannery B. P., 2007, Numerical Recipes 3rd Edition: The Art of Scientific Computing. Cambridge University Press 33, 35

Rybicki G. B., Lightman A. P., 1979, Radiative processes in astrophysics. Wiley, New York 31

Stroe A., Harwood J. J., Hardcastle M. J., Rttgering H. J. A., 2014, MNRAS, 445, 1213 12, 33, 38

Tribble P., 1993, MNRAS, 261, 57 32