RFeditor

User Guide

V3.7.1

Xiaotao Yang, Gary Pavlis, Yinzhi Wang

Indiana University

March 2016

bad_traces.eps

# **Table of Content**

**1. Introduction** 1

1.1 Download 1

1.2 Citation 1

1.3 Support 1

**2. Installation** 1

2.1 Operation Systems 1

2.2 Library dependency 2

2.3 Configure Antelope localmake 2

2.4 Install 2

**3. Data Preparation** 3

**4. Command Line Options** 3

4.1 Running modes 3

4.2 Other options 3

**5. Parameter File** 3

5.1 Global parameters driving the program 3

5.2 Parameters driving the QC in GUI-mode 6

5.3 Parameters driving the QC in Auto-mode 7

5.4 Other optional parameters 8

5.4.1 First arrival detection 8

5.4.2 Metadata lists for input and output tables 9

5.4.3 Keys of the deconvolution attributes 9

5.4.4 Plotting window parameters 9

**6. QC in GUI-Mode** 11

**7. QC in Auto-Mode** 11

**8. References** 11

# **1. Introduction**

In this file, we use symbol ‘$ command’ to denote terminal input. We use $ANTELOPE to denote for the Antelope root path on your computer. We use RFEROOT to the path where you store the RFeditor source package.

## 1.1 Download

The package is available at: <https://github.com/xtyangpsp/RFeditor>. Assumming you have downloaded the package and saved it to: /Users/myhome/SOFT/src/RFeditor, this path is referred to as RFEROOT in this user guide. Under this directory, there are four subfolders:

(1) libtreditoperator

This directory contains the lib needed by RFeditor editing procedures. However, the library files are currently embedded in RFeditor core codes. Thus compiling RFeditor does not require the compiling of this library. This library can be used by other utilities or extensional programs.

(2) RFeditor\_core

This folder contains the RFeditor core source code.

(3) Utilities

This folder includes the utilities related to the use of *RFeditor*.

(4) Docs

This folder contains this user guide and other related materials.

## 1.2 Citation

Reference for the method: X. Yang, G. Pavlis, Y. Wang (in revision). A Quality Control Method for Teleseismic P-Wave Receiver Functions, Bulletin of the Seismological Society of America

## 1.3 Support

Support is provided through GitHub platform. You can crease an issue or ask questions at: <https://github.com/xtyangpsp/RFeditor>. Or you can send emails regarding questions to: <http://xtyang@indiana.edu>.

# **2. Installation**

This program operates based on an Antelope Datascope database (version 5.5 and later). Please make sure that Antelope is working properly before installing this program. To update contrib, under your $ANTELOPE/contrib, type in terminal:

$ git pull

This will update your contrib package. After it finishes, recompile the updated programs.

## 2.1 Operation Systems

This program has been fully tested under Mac OSX Mountain Lion (10.8), Mavericks (10.9), and Yosemite (10.10). For other linux-based systems, it should be working but please report any issues at: https://github.com/xtyangpsp/RFeditor.

## 2.2 Library dependency

The following libraries are required in order to compile RFeditor (current version 3.0):

(1) libseisw:

Seismic widget library for plotting seismic traces as wiggles. It is part of the Antelope contrib package.

(2) boost (http://www.boost.org):

This is a C++ library. On Mac OSX, you can install it from fink.

$fink list boost

$fink install boostpackagename

(3) xmotif: xwindow libraries;

On Mac OSX, you can install motif from fink. In terminal, type:

$fink list motif

Then, type:

$fink install motifpackagename

Or, you can compile the source code by yourself. This package/source code could be downloaded from: http://motif.ics.com and the link there.

(4) libseispp:

This library is released along with the Antelope contrib package. Once you successfully compiled contrib software, libseispp should be already available.

## 2.3 Configure Antelope localmake

After successfully installed boost and xmotif libraries, run in terminal:

$localmake\_config

This will lead you to the interface where you can enable BOOST and XMOTIF capabilities.

## 2.4 Install

Once you have all of the required libraries installed and properly configured, in terminal, type:

$make

If the above compiling procedure is successful, run:

$make install

This deposit tredit and decon tables to $ANTELOPE/contrib/data/css3.0.ext/, deposit RFeditor executable to $ANTELOPE/contrib/bin, and deposity RFeditor.pf to $ANTELOPE/contrib/data/pf/. Please make sure you have the permission to write, read, and execute programs!

If you the above compiling went through, congratulations, RFeditor is available to you.

# **3. Data Preparation**

# **4. Command Line Options**

You can start RFeditor from the terminal by simply typing the program name: RFeditor. This will give you a brief usage information. For example:

$RFeditor

< version v3.7.0.5 > 3/15/2016

RFeditor dbin dbout [-d outdir][-tredit filename][-rm][-go][-continue]

[-fa fa\_filename][-pf pffile][-laststa xx][-ss subset\_condition][-v|V][-h|H]

\*\* Use -h to print out detailed explanations on the options.

However, to start the program on an Antelope database, it requires two arguments: the input database and the output database, where the edited data will be stored.

## 4.1 Running modes

*RFeditor* has two running modes: interactive mode with Graphical User Interface (GUI-mode) and automated mode (Auto-mode). GUI-mode allows the user to interactively work on each station gather, select quality control procedures and customize parameters for each metric. For a large dataset, we recommend this mode for testing parameters before running in Auto-mode.

By default, the program starts in GUI-mode. To run the program in Auto-mode, use option: --gui-off or -go.

## 4.2 Other options

# **5. Parameter File**

The following are main parameters driving the program (see RFeditor.pf in RFEROOT for example parameter values.

## 5.1 Global parameters driving the program

minimum\_number\_receiver\_functions

Stations with number of events less than this will be skipped automatically with a log message.

FA\_reference\_time

First arrival time shift for plot (default displays First Arrival at time 0). Signs: +, shift to positive time axis; -, shift left to negative time axis. When use\_arrival\_data is set to false, this is the FA time in the data and the trace will be plotted starting from 0 seconds.

use\_arrival\_data

This is a Boolean parameter (true/false). If this is true, the arrival and assoc tables must be provided and will be used in converting the time frame from absolute to relative.

use\_decon\_in\_editing

This is a Boolean parameter (true/false). If this is true, the decon table must be provided. This table is generated by the program associated with the Generalized Iterative Deconvolution method by Wang and Pavlis (2016). When this parameter is set to true, the QC process will enable deconvolution attributes related procedures.

use\_netmag\_table true

This is a Boolean parameter (true/false). If this is true, the netmag table must be provided. The program will read in magnitude information from this table. Currently, magnitude information can be used in sorting the receiver functions (a station gather).

radial\_channel\_key

transverse\_channel\_key

vertical\_channel\_key

The above three parameters are channel codes for the three component of the receiver function.

no\_vertical\_data

This is a Boolean parameter (true/false). If this is true, the program will only process radial and transverse data. If your data includes vertical data, this parameter is not required. The default value is false. The program only tries to read this parameter after it fails to read the vertical\_channel\_key from the parameter file.

edit\_on\_channel

This parameter (radial, transverse, or vertical) gives the option to choose which component to edit.

use\_wfdisc\_in

This is a Boolean parameter (true/false). If this is true, the program will read waveform data following wfdisc table. Otherwise, wfprocess table is used.

apply\_prefilter

This is a Boolean parameter (true/false). If this is true, the receiver function data will be filtered before processing and plotting (in GUI-mode). We recommend turn the prefilter on for pure-spike RF data. Otherwise, QC processing will not be working correctly.

wavelet\_type

Available types: filter, gaussian, ricker. Ricker wavelet should NOT be used for the purpose of QC.

filter

Only if wavelet\_type is filter, the filter parameters should be specified here. For example, a Butterworth band-pass filter with 2 poles: BW 0.2 2 2 2. Ignore this parameter if the wavelet\_type is set to gaussian or ricker.

data\_sample\_interval

wavelet\_width\_parameter

wavelet\_length

wavelet\_normalization\_method

The above four parameters are needed by gaussian or ricker wavelet\_type. These are ignored if the wavelet\_type is filter. The data\_sample\_interval is the sample interval in time domain, e.g., 0.025. The wavelet\_width\_parameter is the width of the wavelet. It is the sigma for Gaussian wavelet and the central frequency for Ricker wavelet. The wavelet\_length is length of the wavelet. We suggest that the wavelet is at least 3 times the width parameter in time domain. wavelet\_normalization\_method is recommended to be PEAK.

save\_wfdisc\_table

This is a Boolean parameter (true/false). If this is true, when reading input from wfprocess table, a wfdisc table will be generated containing the receiver functions after QC. The program will use channel keys define above for different channels. This parameter will be ignored if reading input waveform from wfdisc (i.e., use\_wfdisc\_in is true). If use\_wfdisc\_in is false and this parameter is true, then the program saves the output to both wfdisc and wfprocess tables.

save\_wfprocess\_table

This is a Boolean parameter (true/false). If this is true, when reading input from wfdisc table, a wfprocess table will be generated containing the receiver functions after QC. This parameter will be ignored if reading input waveform from wfprocess (i.e., use\_wfdisc\_in is false), when wfprocess table will be saved by default.

save\_3C\_data

This is a Boolean parameter (true/false). It is applicable only if the input is from wfprocess table and with 3c datatype, which means that each row in wfprocess table is a three-component seismogram. If this is true, the waveforms after QC will be saved in 3c datatype. datatype is an entry of the wfprocess and wfdisc table.

save\_decon\_table

This is a Boolean parameter (true/false). When this is true, the decon table will be saved after QC. However, ONLY one component of the decon table will be saved. This component is specified by edit\_on\_channel and the associated channel key will be used in saving the table. This parameter will be ignored if use\_decon\_in\_editing is false.

save\_vertical\_channel

This is a Boolean parameter (true/false). When it is true, vertical component of the three component seismogram will be saved after QC. This is automatically set to true if the input datatype is 3c (i.e., ThreeComponentSeismogram). We use this parameter because there are some imaging programs that only use radial and transverse components.

save\_metadata\_only

This is a Boolean parameter (true/false). When it is true, the program only saves database tables but not the real waveform data. This works for the case when the same type of input and out tables are used. For example, both use\_wfdisc\_in and save\_wfdisc\_table are true. When use\_wfdisc\_in is false, which means use wfprocess as input table, the datatype is 3c, and save\_wfdisc\_table is true, the program currently is unable to handle this combination and will throw errors and exist.

save\_filtered\_data

The user could choose to save the filtered data by tuning this on (or set to true). If this is true, the data after QC will be filtered before saving using the filter defined above by apply\_prefilter and other related parameters.

output\_dfile\_base

Base of the output data file name, e.g., RFedited. The file will be named as: RFedited\_KF28.R for radial, \*.T for transverse, and \*.Z for vertical. When saving 3C data, the file extension will be \*.3C for wfprocess table, and \*.w for wfdisc table.

## 5.2 Parameters driving the QC in GUI-mode

The parameter array, gui\_edit\_parameters &Arr{ }, contains all of the parameters used by GUI-mode RFeditor.

#stacking window params for robustSNR stacking.

#stacktype RobustSNR

robust\_window\_start -1

robust\_window\_end 10

NFA\_tolerance\_TW\_start -2

NFA\_tolerance\_TW\_end 5

PCoda\_search\_TW\_start 5

PCoda\_search\_TW\_end 35.0

PCoda\_grow\_tolerance 0.0

max\_trace\_abs\_amplitude 100 #true amplitude (as stored in the data).

CodaCA\_search\_TW\_start 2.0

CodaCA\_search\_TW\_end 20.0

CodaCA\_tolerance\_twin\_length 5 #recommend: 5\*(filter width in time-domain).

RefXcor\_search\_TW\_start -1

RefXcor\_search\_TW\_end 10

#decon parameter threshold: default values.

niteration\_min 20

niteration\_max 1000

nspike\_min 20

nspike\_max 1000

epsilon\_min 0.0

epsilon\_max 50.0

peakamp\_min 0.001

peakamp\_max 1

averamp\_min 0.0

averamp\_max 1

rawsnr\_min 1

rawsnr\_max 1000

## 5.3 Parameters driving the QC in Auto-mode

apply\_klat false

apply\_decon\_ALL true

apply\_kdnitn false

apply\_kdnspike false

apply\_kdepsilon false

apply\_kdpkamp false

apply\_kdavamp false

apply\_kdsnr false

apply\_kldsi true

apply\_knfa true

apply\_kgpc true

apply\_kca true

apply\_klsw true

apply\_klxcor true

apply\_klrfqi true

NFA\_tolerance\_TW\_start -2

NFA\_tolerance\_TW\_end 5

PCoda\_search\_TW\_start 5.0

PCoda\_search\_TW\_end 35.0

CodaCA\_search\_TW\_start 2.0

CodaCA\_search\_TW\_end 20.0

RefXcor\_search\_TW\_start -1

RefXcor\_search\_TW\_end 10

max\_trace\_abs\_amplitude 100

This is the threshold for maximum true amplitude (as stored in the data).

CodaCA\_tolerance\_twin\_length 5

PCoda\_grow\_tolerance 0.0

RFQI\_weigth\_stackweight 0.2

RFQI\_weigth\_refxcorcoe 0.3

RFQI\_weigth\_successindex 0.5

rfqi\_min 0.6

niteration\_min 20

niteration\_max 1000

nspike\_min 20

nspike\_max 1000

epsilon\_min 0.0

epsilon\_max 50.0

peakamp\_min 0.001

peakamp\_max 1

averamp\_min 0.0

averamp\_max 1

rawsnr\_min 1

rawsnr\_max 1000

stackweight\_min 0.2

xcorcoe\_min 0.5

dsi\_min 0.3

stacktype RobustSNR

robust\_window\_start -1

robust\_window\_end 10

## 5.4 Other optional parameters

### 5.4.1 First arrival detection

First Arrival (FA) detection parameters are not required if FA detection is not turned on. Use -fa option when running RFeditor to turn it on. Here we list these parameters and the default values. The user can change the default values by specifying new values in the parameter file.

FA\_sensitivity 10e-4

This is the sensitivity in amplitude: turn on detection only if the amplitude is above this value. this is used in TraceEditOperator object.

FA\_detect\_length 0.8

FA\_search\_TW\_start -5

FA\_search\_TW\_end 5

Time window length for FA detection. The program will only detect FA within this window specified by the start and end time stamps. When detecting FAs, the moving window length is specified the the FA\_detect\_length parameter above. The program searches for FA within this length after first non-zero values (>=FA\_sensitivity). The empirical length value for reference: >=4\*gaussian\_sigma for gaussian or 1.5\*ricker\_side\_lope\_distance for ricker.

data\_shaping\_wavelet\_type GAUSSIAN

This is referenced only when detecting the first arrivals. This is the wavelet type used to generate the receiver functions, which is input of the RFeditor program.

Use RICKER if ricker was used in either deconvolution or the pre-filtering process.

There are three available types: SPIKE, GAUSSIAN, RICKER (CASE SENSITIVE). This parameter is read-in just once when the TraceEditOperator object is initiated.

### 5.4.2 Metadata lists for input and output tables

In parameter file, metadata lists are specified by a table of metadata tags/attributes and metadata types. In the current version of RFeditor, it uses built-in metadata lists unless the user specifies them in the parameter file.

mdlist\_ensemble

Metadata list read in for each time series ensemble (station gather).

mdlist\_wfdisc\_in

Metadata list read in for each receiver function trace when using wfdisc table as input.

mdlist\_wfdisc\_out

Metadata list of the output wfdisc table.

mdlist\_wfprocess\_in

Metadata list read in for each receiver function trace when using wfprocess table as input. Adjust the lists of attributes when using decon table, arrival data or netmag table.

mdlist\_wfprocess\_out

Metadata list of the output wfprocess table.

### 5.4.3 Keys of the deconvolution attributes

The following are default values for decon keys. If you want to use other values, please make sure they are consistent with those in the mdlist\_wfprocess\_in metadata list/table when use\_decon\_in\_editing is true. Those are defined in the parameter arrays, gui\_edit\_parameters &Arr{} and auto\_edit\_parameters &Arr{}.

Defaults definitions:

decon\_nspike\_key decon.nspike

decon\_rawsnr\_key decon.rawsnr

decon\_averamp\_key decon.averamp

decon\_epsilon\_key decon.epsilon

decon\_niteration\_key decon.niteration

decon\_peakamp\_key decon.peakamp

### 5.4.4 Plotting window parameters

Parameters for the plotting window shown here are the built-in default values. PLEASE DO NOT change them unless necessary! You can specify the values for these parameters in the parameter array, gui\_edit\_parameters &Arr{}. Here are list the default values for all of these parameters.

Default values:

SUVariableArea\_grey\_value 1

VariableArea true

WiggleTrace true

blabel Data Window

blabel2 Data Window

clip\_data true

clip\_percent 99.5

clip\_wiggle\_traces false

d1num 0.0

d2num 0.0

default\_curve\_color black

editing\_mode single\_trace

f1num 0.0

f2num 0.0

first\_trace\_offset 0.0

grid1 1

grid2 1

gridcolor blue

hbox 5000

interpolate true

label1 time

label2 index

labelcolor blue

labelfont Rom14

labelsize 18.0

n1tic 5

n2tic 1

plot\_file\_name SeismicPlot.ps

style normal

time\_axis\_grid\_type solid

time\_scaling auto

title Receiver Function Data

titlecolor red

titlefont Rom22

titlesize 36.0

trace\_axis\_attribute assoc.delta

trace\_axis\_grid\_type none

trace\_axis\_scaling auto

trace\_spacing 1.0

trim\_gap\_edges true

use\_variable\_trace\_spacing false

verbose true

wbox 950

windowtitle RFeditor

x1beg 0.0

x1end 120.0

x2beg 0.0

x2end 24.0

xbox 50

xcur 1.0

ybox 50

beam\_hbox 250

beam\_clip\_data false

beam\_trace\_spacing 1.0

beam\_xcur 1.0

beam\_trace\_axis\_scaling auto

# **6. QC in GUI-Mode**

# **7. QC in Auto-Mode**

# **8. References**