Manual of conti2d

Zhikui Guo, Chunhui Tao

Key Laboratory of Submarine Geosciences, SOA, Second Institute of Oceanography, MNR, Hangzhou

310012, China

1. Description

Conti2d is a free source and cross-platform C++ program to calculate upward and downward continuation of potential field data. The algorithm is implemented in spatial domain. Every file can be downloaded from github repository. There are two branches, one is master and the other is win10. The files in win10 branch are for windows users and the master branch is for the other system users, e.g. Mac OS and Linux. In addition, we also developed a GUI (Graphical User Interface) version of conti2d based on Gmsh.

2. Windows

2.1 Download

The source code files and precompiled executable file can be downloaded from win10 branch: https://github.com/zguoch/conti2d/tree/win10, see figure 1.

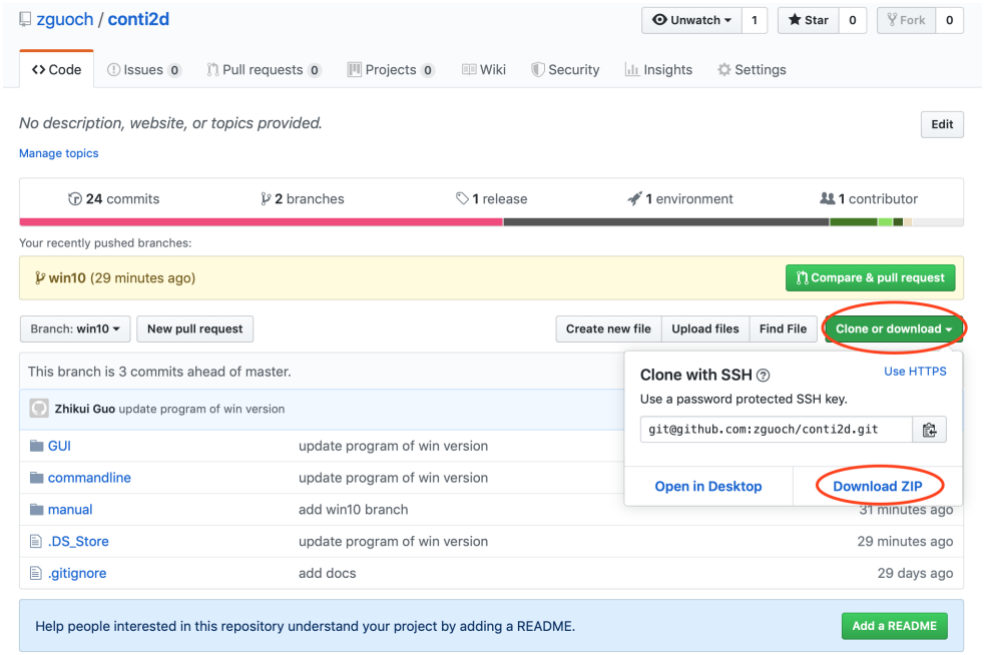

Figure 1. Snapshot of download files from win10 branch

The folder structure and explanation are shown in figure 2.

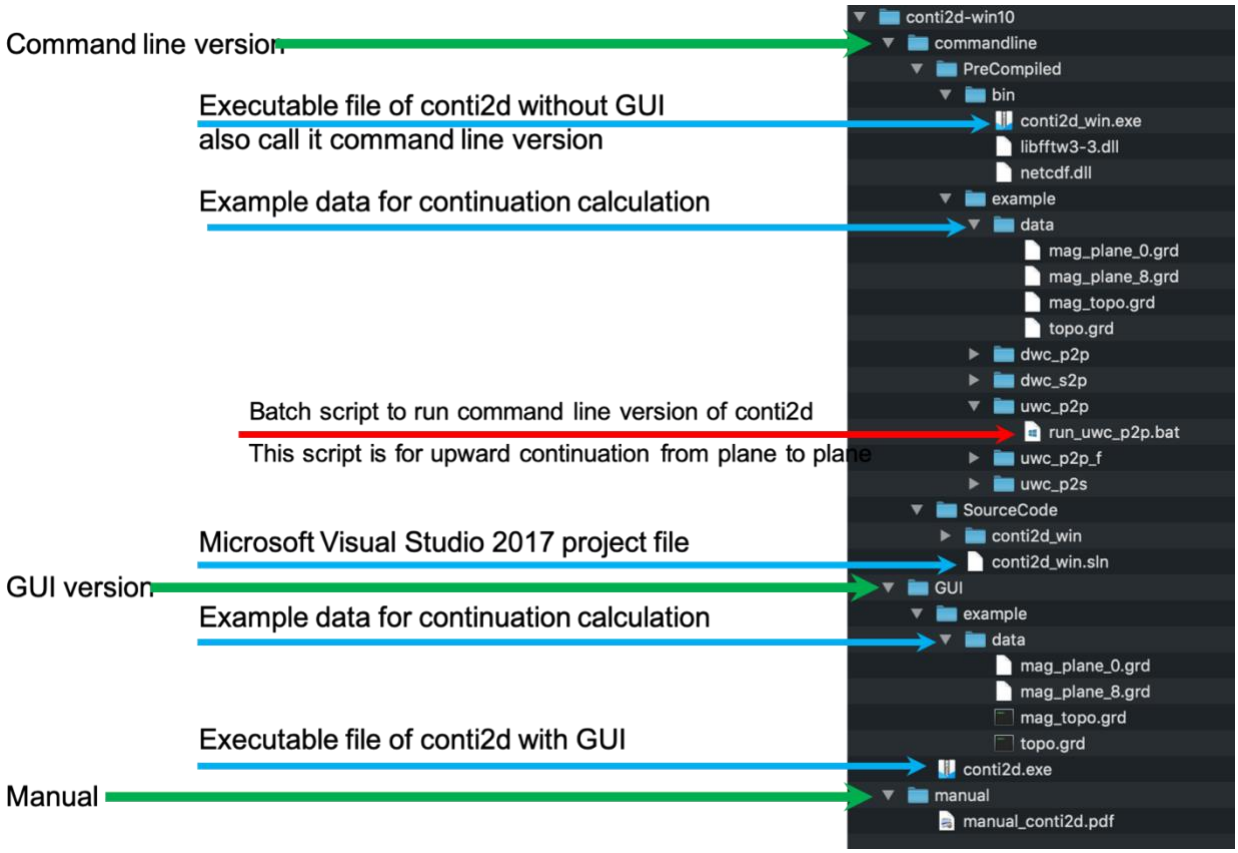

Figure 2. Folder layout and explanations

2.2 Build

If you can run the GUI version (the file of GUI/conti2d.exe) or the command line version (the file of commandline/Precompiled/bin/conti2d_win.exe)(see figure 2 for file location), please skip section 2.2.

The compilation of command line version is very simple, but you have to install Microsoft Visual Studio 2017 (Community version is free for academic purpose) on your system. I have configured everything into a Microsoft-Visual-Studio-2017 project file, which is located in commandline/Sourcecode/conti2d_win.sln (see figure 2). You just double click conti2d_win.sln file and rebuild it, then you can get the new executable file compatible with your system. This file will be at this location, commandline/Sourcecode/x64/Release/conti2d_win.exe.

The compilation of GUI version is much more complex, because the project is configured by cmake and some other libraries(i.e. zlib, fltk) must be installed firstly. This will be introduced in the section 3.

2.3 Usage

2.3.1 GUI version

The GUI version is easy to use, just click the buttons using mouse and type parameters through dialog. The tooltips for all the parameters and buttons are embedded in the software, it is easy to understand the meaning when your mouse points to the items. For example download continuation from an undulant surface to a horizontal plane, see figure 3. It only needs four parameters, which are listed as green text in figure 3.

The first step is clicking the module buttons on the left and a parameter setting dialog pop up, there are four mudules (1) Upward->Plane to Plane, means upward continuation from the observation plane to another plane above. (2) Upward->Plane to Surface, means upward continuation from the observation plane to an undulant surface above. (3) Downward->Plane to Plane, means downward continuation from the observation plane to another plane below. (4) Downward->Surface to Plane, means downward continuation from the observation surface to a horizontal plane below. The second step is setting each parameters by following the tooltips, and

then click Run button to start calculating. To make this manual much more clear, I upload a tutorial video to YouTube, the link is https://youtu.be/jQcnD3rnlkg.

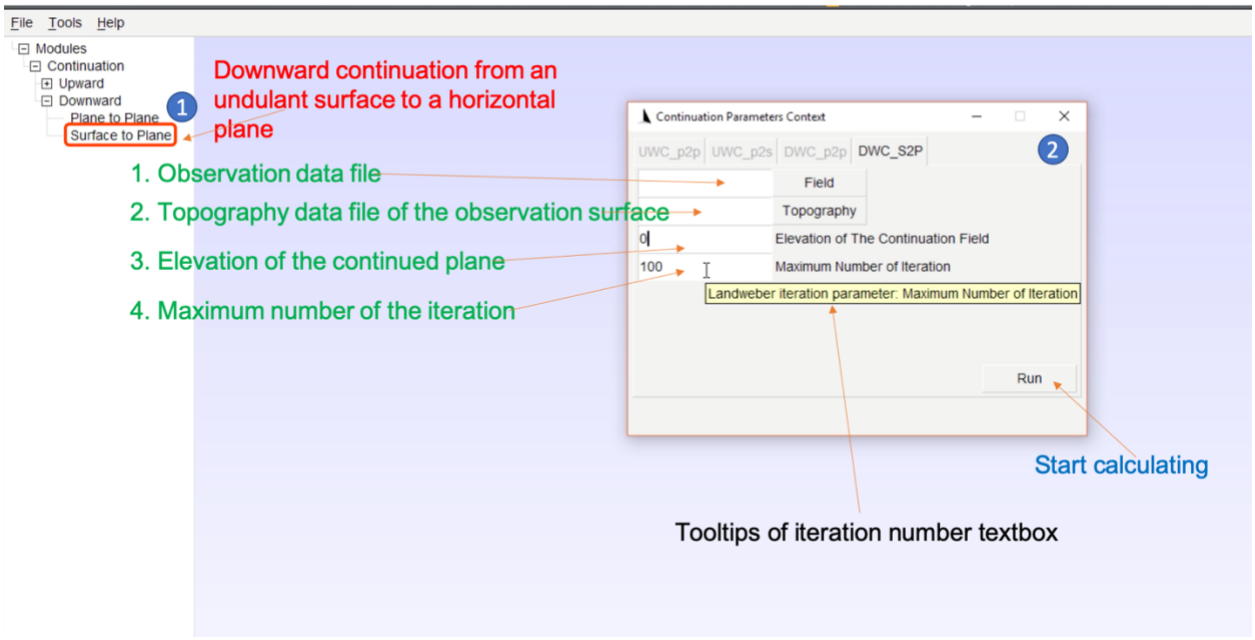

Figure 3. Snapshot of parameter setting of downward continuation from surface to plane for the GUI version.

The calculation result can be saved by clicking menu of File->Save Results.

For the downward continuation, the temporary results of each iteration step are saved in the same directory, the data for smooth-fitting curve is saved in file of LandweberIteration_log.txt in the same folder. The first column is iteration number, second column is $\left\|U_0 - KU^t\right\|_2 / \left\|U_0\right\|_2$ representing fitting, the third column is $\left\|U^t\right\|_2$ and the fourth column is $\left\|grad(U^t)\right\|_2$ which is formulated in equation 11 in the manuscript, representing smooth. The optimal iteration number can be determined by fitting-smooth curve, i.e. see figure 10 in the manuscript.

2.3.2 Command line version

The command line version provides six arguments, they -G, -H, -T, -D, -t and -f, which are presented in green text below. The usage of arguments is similar with the commonly used software gmt.

Synopsis

conti2d Inputfile **-G**outputfile -EextBoundNum **-H**level_outputdata|grdfile **-T**level_inputdata|grdfile **-D**[+Llandweber_par] **-t**threads **-f**

Note: No space is allowed between the option flag and the associated arguments.

1) Upward continuation from plane to plane

Inputfile

Names of data file to be continued. The supported input file format is ASCII Golden Software Surfer 6 format (see example data files)

-G*outputfile*

Output file name. The supported file format are ASCII Golden Software Surfer format, netCDF, vtk, which are identified by the extension name .grd, .nc and .vtk respectively. For example, -Gout.vtk means the output grid data file in vtk format, which can be viewed by Paraview. The .nc file can also be view by Paraview. The .grd and .nc file can be plotted by Golden Software Surfer and GMT command of grdimage.

-H*height*

A value represents height of upward continuation.

Example: upward continuation from plane of z=0 to plane of z=8

```
conti2d mag_plane_0.grd -Gmag_plane_uwc_8.vtk -H8
```

See complete batch script file in the example folder.

2) Upward continuation from plane to surface

Inputfile

Names of data file to be continued. The supported input file format is ASCII Golden Software Surfer format (see example data files)

-G*outputfile*

Output file name. The supported file format are ASCII Golden Software Surfer format, netCDF, vtk, which are identified by the extension name .grd, .nc and .vtk respectively. For example, -Gout.vtk means the output grid data file in vtk format, which can be viewed by Paraview. The .nc file can also be view by Paraview. The .grd and .nc file can be plotted by Golden Software Surfer and GMT command of grdimage.

-T*elevation_observation_field*

A value represents elevation of the observation field.

-H*topography_continued_surface*

A grid data file represents the continued surface. Note that the grid of *Inputfile* and *topography_continued_surface* must be the same.

Example: upward continuation from plane of z=0 to a surface of topo.grd

```
conti2d mag_plane_0.grd -Gmag_plane_uwc_p2s.vtk   -T0 -Htopo.grd
```

See complete batch script file in the example folder.

3) Downward continuation from plane to plane

Inputfile

Names of data file to be continued. The supported input file format is ASCII Golden Software Surfer format (see example data file in the attachement)

-G*outputfile*

Output file name. The supported file format are ASCII Golden Software Surfer format, netCDF, vtk, which are identified by the extension name .grd, .nc and .vtk respectively. For example, -Gout.vtk means the output grid data file in vtk format, which can be viewed by Paraview. The .nc file can also be view by Paraview. The .grd and .nc file can be plotted by Golden Software Surfer and GMT command of grdimage.

-H*elevation_continued_plane*

A value represents elevation of the downward continued plane below.

-T*elevation_observation_plane*

A value represents elevation of the observation plane.

-D+L*iteration_number*

D means downward continuation. L means Landweber method. *iteration_number* is the maximum iteration number.

Optional Arguments

-t*threads*

A integer (>1) represents how many threads do you want to use. The default value is the maximum threads of your computer.

Example: downward continuation from plane of z=8 m to a plane of z=0 m

conti2d mag_plane_8.grd -Gmag_plane_dwc_p2p.vtk -T8 -H0 -E5 -D+L500

See complete batch script file in the example folder.

4) Downward continuation from surface to plane

Inputfile

Names of data file to be continued. The supported input file format is ASCII Golden Software Surfer format (see example data file in the attachement)

-G*outputfile*

Output file name. The supported file format are ASCII Golden Software Surfer format, netCDF, vtk, which are identified by the extension name .grd, .nc and .vtk respectively. For example, -Gout.vtk means the output grid data file in vtk format, which can be viewed by Paraview. The .nc file can also be view by Paraview. The .grd and .nc file can be plotted by Golden Software Surfer and GMT command of grdimage.

-H*elevation_continued_plane*

A value represents elevation of the continued plane below.

-T*topography_observation_surface*

File name of topography data of the observation surface.

-D+L*iteration_number*

D means downward continuation. L means Landweber method. *iteration_number* is the maximum iteration number.

Example: downward continuation from surface of topo.grd to a plane of z=0 m

conti2d mag_plane_8.grd -Gmag_plane_dwc_p2p.vtk -Ttopo.grd -H0 -E5 -D+L500 -t8

See complete batch script file in the example folder.

3. Linux and Mac OS

On Linux and Mac OS, the download and build process are much more simple than on the

windows system, just use several command lines. Note that all the following process is based on the master branch in the github repository.

3.1 Download

```
git clone git@github.com:zguoch/conti2d.git
```

3.2 Build

1) Command line version

```
cd conti2d
make
```

2) GUI version

```
cd conti2d/GUI
mkdir build
cd build
cmake ..
make
```