

StagLab 3

Geodynamic Diagnostics and Scientific Visualisation for Geodynamic Models

DOI 10.5281/zenodo.1199038

Author

- **Fabio Crameri** - [contact](#)

Various [contributors](#) to this project are mentioned below.

Prerequisites

StagLab necessitates a working version of **MatLab 2014b or later**.

StagLab works best with the latest version of MatLab as earlier versions might disable multiple StagLab features and might cause problems as compatibility is not maintained carefully any longer.

Installing

- To install StagLab, simply execute the included ***f_INSTALL***. This can be done in the MatLab terminal by typing:

```
cd <yourPath>/StagLab3
f_INSTALL
```

Alternatively, add all StagLab files manually to the MatLab search path (in MatLab go to: *HOME > Set Path > Add With Subfolders*).

- It is best practise to delete old StagLab versions.

StagLab removes, however, file duplicates from the MatLab search path and so prevents confusion with old files.

Testing

- To test StagLab on your system, simply execute the included ***f_TEST***. This can be done in the MatLab terminal by typing:

```
cd <yourPath>/StagLab3
f_TEST
```

This automated test performs various core tasks of StagLab and produces a suite of test figures that are saved to *StagLab > Examples > ExampleFigures*.

Running

- StagLab is run through parfiles (see folder *Parfiles*). Use one of the parfiles included (e.g., ***ParStagLab2D***) to set your parameters and to run one of the main StagLab Apps (*STAGplot* for parameter fields, *STAGrprof* for radial profiles, *STAGtimedat* for time evolutions). See ***f_Defaults***, ***f_DefaultsRprof***, or ***f_DefaultsTimedat*** for all available options available with the corresponding parfile.

	Parameter Fields	Radial Profiles	Temporal Graphs
Execution File	<i>ParStagLab2D</i> <i>ParStagLab3D</i> <i>ParStagLabYY</i>	<i>ParStagLabRprof</i>	<i>ParStagLabTimedat</i>
Defaults	f_Defaults	f_DefaultsRprof	f_DefaultsTimedat
Routine	STAGplot	STAGrprof	STAGtimedat

You can run your parfile from any directory you like.

You will always be able to re-use your old parfiles to run newer versions of StagLab.

- Adjust file name, number and directory with:

```
IN.Name           = {'test'};
IN.Number         = [1];
IN.Folder         = {'/work/stagyy/'};
```

TIP: Given the above *IN.Folder*, StagLab checks automatically also for the following folder-structures to read:

```
/work/stagyy/+op/<fileToRead>
/work/stagyy/+op/<filename>/<fileToRead>
```

and write:

```
/work/stagyy/+im/<fileToSave>
/work/stagyy/+im/<filename>/<fileToSave>
```

TIP: It is possible to plot or compare multiple files in the same figure by simply adding another file name. *IN.Name* controls which and how many files are plotted. To plot three different files all for the first output number:

```
IN.Name           = { 'test1' 'test2' 'test3' };
IN.Number         = [ 1 ];
IN.Folder         = { '/folder1/' '/folder2/' '/folder3/' };
```

To plot multiple time steps of one single model:

```
IN.Name           = { 'test1' 'test1' 'test1' };
IN.Number         = [ 1 2 3 ];
IN.Folder         = { '/folder_test1/' };
```

If there is just one entry for either *IN.Number* or *IN.Folder*, it will take the same entry for all files specified in *IN.Name*.

- Adjust the dimensional parameters in the parfile for correct dimensionalisation according to ***f_Dimensions***.

```
IN.Parameter      = [ 11 ];
```

StagLab saves the publication-ready figures and movies, if:

```
SAVE.Figure       = logical(1);
SAVE.Movie        = logical(1);
```

To specify a certain write directory change the default:

```
SAVE.writeDirectory = 'auto';
```

to e.g.:

```
SAVE.writeDirectory = '/work/stagyy/';
```

NOTE: Preparing Fluidity Output

To make readable by StagLab, the original Fluidity output needs to be converted to a .csv file using e.g., Paraview. Adjust StagLab's ***f_readFluidity*** to the specific details of the .csv file.

Acknowledging StagLab

- Please acknowledge the free use of the StagLab or any of its routines.

Use for example:

"The Geodynamic diagnostics and scientific visualisation software StagLab (Crameri 2017; Crameri 2018) is used in this study."

*Crameri, Fabio. (2017, December 7). StagLab 3.0 (Version 3.0.2). Zenodo.
<http://doi.org/10.5281/zenodo.1199038>*

Crameri, F. (2018, in open review), Geodynamic diagnostics, scientific visualisation and StagLab 3.0, Geosci. Model Dev. Discuss., doi:10.5194/gmd-2017-328

Published Studies Using StagLab

- Crameri and Lithgow-Bertelloni (2017), Tectonophysics, [doi:10.1016/j.tecto.2017.09.013](https://doi.org/10.1016/j.tecto.2017.09.013).
- Crameri et al. (2017), Gcubed, [doi:10.1002/2017GC006821](https://doi.org/10.1002/2017GC006821).
- Crameri and Tackley (2016), PEPS, [doi:10.1186/s40645-016-0103-8](https://doi.org/10.1186/s40645-016-0103-8).
- Crameri and Tackley (2015), JGR, [doi:10.1002/2014JB011664](https://doi.org/10.1002/2014JB011664).
- Crameri and Tackley (2014), JGR, [doi:10.1002/2014JB010939](https://doi.org/10.1002/2014JB010939).
- Crameri et al. (2012), GRL, [doi:10.1029/2011GL050046](https://doi.org/10.1029/2011GL050046).

Contributing

- Direct contributions, bug reports, requests and general questions to [Fabio Crameri](#).

Contributors

External Routines

- `f_readStagYY` originally by **Boris Kaus** to read StagYY's binary output directly into MatLab.
- `f_readFluidity` originally by **Fanny Garel** to read Fluidity's CSV output into MatLab.
- `f_YYtoMap` originally by **Paul Tackley** to produce horizontal maps of fully spherical yinyang data.
- `export_fig` originally by **Oliver Woodford** to add more flexibility to figure saving.
- `flowfun` and `cumsimp` by **Kirill K. Pankratov** to derive the stream function and to perform the Simpson-rule column-wise cumulative summation, respectively.
- `MinVolEllipse` by **Nima Moshtagh** to fit a minimum-volume ellipse around a point cloud.
- `hatchfill2` originally by **Neil Tandon** to fill areas with a specific texture.
- `plotboxpos` by **Kelly Kearney** to return the position of the plot more accurately and reliably.
- a few functions including `equalisecolourmap.m` and `sineramp2.m` by **Peter Kovesi** were used to provide the scientific colour-map diagnostics.

Thanks to...

- **Kiran Chotalia** for her help with debugging the code and additions to STAGrprof.
- **Antioniette Grima** for her help with debugging the code.
- **Enrico Marzotto** for his help with debugging the code.
- **Fanny Garel** for her help with adding compatibility with Fluidity output.
- **Robert Petersen** for his help with deriving an appropriate radius of curvature for the plate bending.
- **Tobias Rolf** for his help with the surface variation histogram plot.
- **Marcel Thielmann** for his help with StagLab's compatibility across different MatLab versions
- **Paul Tackley** for his helpful comments throughout the development of StagLab.

Versioning

StagLab 3

- introducing automated installation and testing
- introducing 2-D mode for 3-D
- introducing analysis mode
- introducing tracer plot
- introducing surface-variation histogram plot
- introducing topography diagnostics
- introducing perceptually-uniform colour schemes
- option to discretize colour maps
- option to set default figure position on screen
- option to shift or flip data horizontally
- support for partial cylindrical geometry
- magnifier support for cylindrical geometry
- additional parameter fields
- additional plate diagnostics
- major improvements to STAGrprof and STAGtimedat
- compatibility with Fluidity output
- refined visual design
- improved file finder
- improved code design
- improved speed
- bug fixes

StagLab 2

- introducing mantle-dynamics diagnostics
- introducing tectonic diagnostics
- introducing topography components (isostatic,residual)
- introducing plot for up- and downwelling

- introducing parameter table
- introducing plot-in-plot mode
- introducing movies
- introducing fAlo
- more parameter fields added
- automatic detection of side-boundary v-condition
- less-disruptive error handling
- cleaner plot design and layout
- improved colormaps
- improved filefinder
- improved display output
- improved stability of design-routines
- improved saving and plotting of tectonic data
- improvements towards convertibility to other geodynamic codes
- bug fixes

StagLab 1

- Combining Apps to StagLab
- supports all available StagYY model geometries
- supports all available StagYY output, including rprof.dat and time.dat
- option added for YinYang horizontal maps
- hot and cold plume tracking added
- first test version of 3-D Cartesian plate boundary tracking implemented
- code speed optimisations: deriving lithosphere thickness
- option added to plot horizontal residual temperature
- option added to plot heat flux
- option added to plot temporal evolution of tectonic parameters
- option added to save figure to specific directory
- improved code design
- improved user friendliness
- bug fixed that led to empty plate sketch plot
- bug fixes and updates to the dimensionalisation

Reference

- Crameri, F. (2018), *Geodynamic diagnostics, scientific visualisation and StagLab 3.0*, *Geosci. Model Dev. Discuss.*, [doi:10.5194/gmd-2017-328](https://doi.org/10.5194/gmd-2017-328)

License

StagLab, its individual subroutines and the included colour-map suite are licensed under a [Creative Commons Attribution 4.0 International License](#).

Copyright (c) 2018, Fabio Crameri All rights reserved.

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

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution

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

