A users guide to flipper (0.8.4)

Mark Bell

July 25, 2015

> To get, install, test and start the flipper application under Python using Pip:
>
> ```
> > python -m pip install flipper --user --upgrade
> > python -m flipper.test
> > python -m flipper.app
> ```

Flipper is a program for computing the action of mapping classes on laminations on a punctured surface using ideal triangulation coordinates. It can decide the Nielsen–Thurston type of a given mapping class and, for pseudo-Anosov mapping classes, construct a layered, veering triangulation of their mapping torus, as described by Agol [2]. Flipper is currently under development and this users guide will be based on flipper 0.7.7.

Flipper can be run as a Python 2, Python 3 or Sage Python module [6]. It has been tested on:

Windows 7	Windows 8.1	Ubuntu 14.04
Python 2.7.8		Python 2.7.8
Python 3.4.2	Python 3.4.2	Python 3.4.2
		Sage 6.5

Some of its features require exact arithmetic. Although these calculations can be done in pure Python, Sage's libraries appear to be significantly faster and so are recommended. For even more speed (~25% more) consider running flipper with the `-O` optimise bytecode option.

1 Getting flipper

Flipper is available on the Python Package Index (PyPI). The preferred method for installing the latest stable release is to use Pip:

```
> python -m pip install flipper --user --upgrade
```

Pip can be installed using `get-pip.py` from `http://pip.readthedocs.org/en/latest/installing.html` and is included in Python 2.7.9 and Python 3.4 by default.

Although not officially supported, flipper can often also be installed using `easy_install`:

```
> python -m easy_install flipper
```

1.1 Optional dependencies

The flipper kernel has no required dependencies. However some of the included subpackages do require additional modules:

- Tkinter: Required for the flipper application. On Ubuntu install using the command:

  ```
  > apt-get install python-tk
  ```

Note that if you are running flipper as a Sage Python module then your Tkinter install may not be recognised and you may see an error such as: `Error: no module named _tkinter`. To fix this install the tcl/tk development library and then rebuild Sage's Python. On Ubuntu you can do this using the commands:

```
> apt-get install tk8.5-dev
> sage -f python
```

- SnapPy [5]: Required for some of the examples and tests. Install using the command:

```
> python -m pip install snappy --user --upgrade
```

1.2 Citing flipper

If you find flipper useful in your research, please consider citing it. A suggested reference is:

```
Mark Bell. flipper (Computer Software).
https://bitbucket.org/Mark_Bell/flipper/, 2013--2014. Version <<version>>.
```

or the BibTeX entry:

```
@Misc{flipper,
  author = {Bell, Mark},
  title = {flipper (Computer Software)},
  howpublished = {\url{https://bitbucket.org/Mark_Bell/flipper/}},
  year = {2013--2014},
  note = {Version <<version number used>>}
}
```

or the BibItem:

```
\bibitem{flipper} Mark bell: \emph{flipper (Computer Software)},
  \url{https://bitbucket.org/Mark_Bell/flipper/}}, (2013--2014),
  Version <<version number used>>
```

1.3 Flipper development version

Although the latest stable release of flipper is available through PyPI, you can get the latest development version of flipper from `https://bitbucket.org/mark_bell/flipper` or straight from the Mercurial [1] repository with the command:

```
> hg clone https://bitbucket.org/mark_bell/flipper
```

To compile flipper use the command:

```
> python setup.py install --user
```

At some point the BitBucket repository may also include precompiled binaries.

1.4 Known issues

- Before version 0.7.1, flipper was listed on PyPI as `flipper-triangulation`. If you have a version older than 0.7.0 you may need to manually remove it before beginning the install / upgrade process.

- There is currently no support for flipper on Mac OS. Additionally there are several known issues with tcl/Tk on Mac OS X, see `https://www.python.org/download/mac/tcltk` for more information.

2 The flipper application

Some basic commands that can be run once flipper has been installed are listed in Table 2.

Command	Effect
‖ > `python -m flipper`	Display information about flipper.
‖ > `python -m flipper.app`	Start the flipper GUI application.
‖ > `python -m flipper.doc`	Open this documentation.
‖ > `python -m flipper.test`	Run the test suite.
‖ > `python -m flipper.example`	List included examples.
‖ > `python -m flipper.profile`	Run the profiling suite.

Table 1: Flipper package commands.

2.1 Getting started

Once installed the flipper application can be started by using the command:

‖ > `python -m flipper.app`

The main window of flipper is shown in Figure 1. It has a canvas for drawing on (1) and a list of known laminations and mapping classes (2).
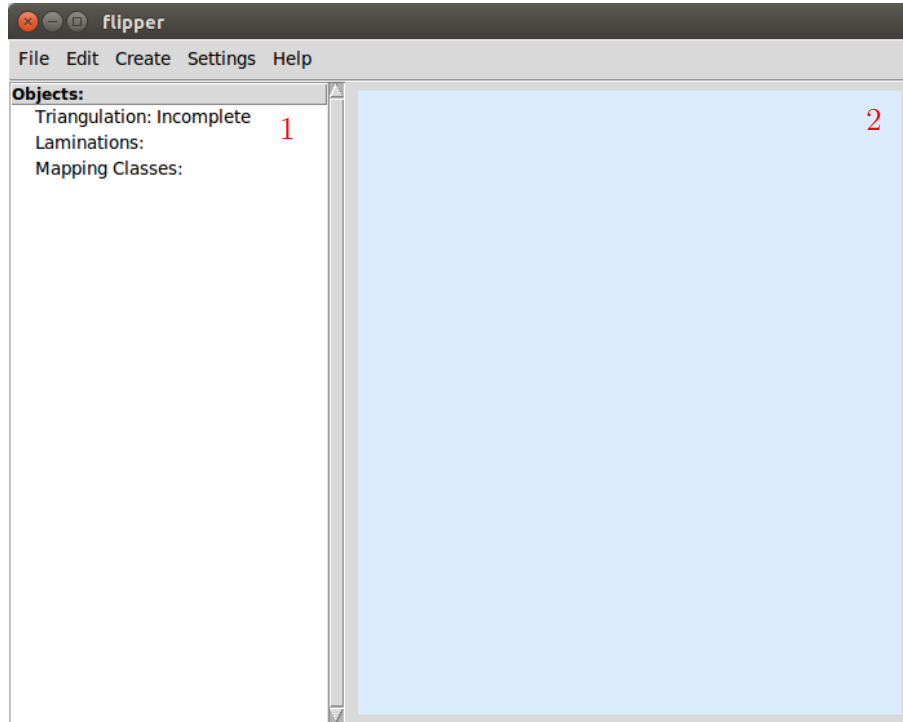

Figure 1: The main flipper window.

Remark 2.1. The currently selected object is highlighted in red. You can cancel your current selection at any time by clicking on the object again, pressing `Escape` or double clicking. Additionally you can delete the currently selected object by pressing either `Delete` or `Backspace`.

2.2 Included Examples

Flipper includes several example triangulations of surfaces. One can be loaded by using the action `File > Open example...`. Flipper includes all of the standard generating sets used by [3] and [4].

From within this menu you can also request flipper to automatically construct a polygon with a desired edge identification. To do this use the option `circular n-gon` and specify the gluings by a string of upper and lower case letters. Each letter of this string corresponds to an edge, ordered anti-clockwise, and an edge with a lower case letter is identified with the edge with the corresponding upper case letter. For example `abAB` gives the once-punctured torus and `xyzXZY` gives the twice punctured torus.

Similarly the option `radial n-gon` creates a polygon with the specified edge identifications but with an additional vertex at its centre.

2.3   Creating a triangulation

To create a triangulation, click on the canvas to create vertices. Click on two in succession to connect them via an edge. You cannot add an edge if it would meet the interior of an existing edge. Flipper automatically adds triangles between any triple of vertices, each pair of which is connected via an edge. Click on two edges, each of which are part of exactly one triangle, in succession to identify them. Clicking on an identified edge will destroy the identification.

The triangulation is *complete* if each edge is either contained in two triangles or is contained in one triangle and is identified with another. Once the triangulation is complete flipper will switch to interpreting clicks as drawing a lamination. You can force flipper to place a vertex or select an edge, even if the triangulation is complete, by holding `Shift` while clicking.

2.4   Adding laminations

Once the surface drawn is complete you can start drawing laminations on it. Click on the canvas outside of the drawn surface to start drawing a lamination. Click on the canvas again to extend it through the current point. Click outside of the drawn surface to finish drawing.

You can remove the last point currently being drawn by either pressing `Delete` or `Backspace`. You should make sure to draw transverse to the underlying triangulation.

The currently drawn lamination can be added to the list of known laminations by using the action `Create > Lamination`.

2.5   Adding mapping classes

There are currently four different ways of creating mapping classes.

- Dehn twist - A Dehn twist about the currently drawn lamination can be created by using the action `Create > Mapping class > Twist`. This can only be done if the lamination is actually a curve.

- Half twist - A half twist about the currently drawn lamination can be created by using the action `Create > Mapping class > Half twist`. This can only be done if the lamination is actually a curve which cuts of a twice marked disk.

- Isometry - An isometry of the underlying triangulation can be created by using the action `Create > Mapping class > Isometry`. The available isometries are listed according to how they permute edges.

- Composition - A composition of existing mapping classes can be created by using the action `Create > Mapping class > Composition`. The composition should be specified by a string of mapping class names separated by periods where necessary. Powers may be specified and the swapcase of a name is synonymous with its inverse mapping class.

Remark. Currently flipper is only capable of computing Dehn twists and half twists about *non-isolating curves*, where every complementary region contains at least one puncture. Hence, for example, half twist on a twice marked surface cannot be computed in the application.

2.6 Object properties

Stored laminations and mapping classes appear in the object list. Clicking on a object will show more information about it and actions involving it.

Mapping classes also include a list of properties. If a property can be computed in polynomial time then it is automatically listed otherwise it will be listed as ?. You can ask flipper to compute any unknown property by double clicking on it. Some properties prevent other actions from being taken and so are listed as x. For example, flipper cannot compute the invariant lamination of a periodic mapping class.

The properties of a mapping class are:

- Order: Infinite / $\mathbb{N}$.

- Type: Periodic / Reducible / Pseudo-Anosov - The Nielsen–Thurston type of this mapping class.

- Invariant lamination - Finds a lamination which is projectively invariant under this mapping class.

- Conjugate to - Determines if this mapping class is conjugate to another.

- Bundle - Constructs a layered, veering triangulation of this mapping classes mapping torus.

References

[1] Mercurial, a distributed source control management tool. Available at `http://mercurial.selenic.com/`. [2]

[2] Ian Agol. Ideal triangulations of pseudo-Anosov mapping tori. In *Topology and geometry in dimension three*, volume 560 of *Contemp. Math.*, pages 1–17. Amer. Math. Soc., Providence, RI, 2011. [1]

[3] Mark C. Bell. Table of Bundles. Available at `https://bitbucket.org/Mark_Bell/bundle-censuses`. [3]

[4] J. C. Cha and C. Livingston. KnotInfo: Table of Knot Invariants. Available at `http://www.indiana.edu/~knotinfo`. [3]

[5] Marc Culler, Nathan M. Dunfield, and Jeffrey R. Weeks. SnapPy, a computer program for studying the topology of 3-manifolds. Available at `http://snappy.computop.org` (20/09/2014). [2]

[6] W. A. Stein et al. *Sage Mathematics Software (Version 6.3.1)*. The Sage Development Team, 2014. `http://www.sagemath.org`. [1]