Hele-Shaw Toolbox User's Guide

Julius Ross and David Witt Nyström

June 2014

1 Summary

The HeleShaw Toolbox is a basic simulation for an approximation of the Hele-Shaw flow in the plane. That is, it simulates a Hele-Shaw flow of a polygon, obtained by injecting fluid at a single point (by default the injection point is the origin) at unit speed in a medium with possibly varying permeability. The three main functions are:

`heleshawflow` - run a Heleshaw flow with the specified initial condition and parameters

`heleshawplot` - plot a Heleshaw flow at a given time step

`heleshawmovie` - create a movie of a Heleshaw flow

Further documentation can be found in the scripts themselves.

2 Prerequisites

The toolbox was written in MATLAB[1] version R2014a. To use it, add the directory to your MATLAB path. The package depends on the Schwarz-Christoffel Toolbox [2] which must also be installed to the MATLAB path.

3 Getting Started Examples

The following example is sufficient to get started. The initial data needs to be a counterclockwise polygon whose vertices are specified by a column with complex entries giving the vertices of a polygon.

```
» p=transpose([1+i -1+i -1-i 1-i])
```

You can then run a HeleShaw flow starting with this initial condition.

```
» H = heleshawflow(p)
```

[1]MATLAB is a registered trademark of the MathWorks, Inc.

This will run a single step of the heleshawflow starting with `p` as an initial condition and using the default parameters. To get something more interesting try instead:

```
» H = heleshawflow(p,'numberofsteps',10)

H =

  heleshaw with properties:

          vertices: [4x12 double]
     regularpoints: [8x12 double]
    conformalradii: [1x12 double]
           density: @(x)1
          timestep: 0.1
```

The most important part of the output is `H.vertices`, which in this case a 4x11 matrix. The n-th column is the polygon of the heleshaw flow at step n.

To visualise the results run

```
» heleshawplot(H)
```

which will give a plot of the last step of this simulation. To see all the steps run:

```
» heleshawplot(H,'steps',[1:11])
```

You can also continue to run an existing heleshawflow simulation, for example

```
» H = heleshawflow(H,'numberofsteps',8)
```

which will result in a further 8 steps being added to the heleshaw object `H`.

To create a movie `» H = heleshawflow(H,'filename','moviefile.avi ')`
`» implay('moviefile.avi ')`
There are various other parameters that can be specified for the flow. Perhaps the most interesting is the possibility to change the density used, which encodes the permeability in the medium (so if $f\colon \mathbb{C} \to \mathbb{R}_{>0}$ is a density, the permeability is encoded by the function $1/f$). For example:

```
>> f=@(x) 1+ abs(real(x)).^2
>> p = bigregularpolygon(8);
>> H1=heleshawflow(p,'density',f,'numberofsteps',10)
>> heleshawplot(H1)
```

Lastly we have a more interesting example:

```
>> f=@(x) 1+ 10*exp(-5*abs(x-1.^2) +
       10*exp(-5*abs(x-1-i)).^2) + 10.*exp(-5*abs(x+1+i).^2)
>> p = bigregularpolygon(32);
>> H2=heleshawflow(p,'density',f,'numberofsteps',10)
>> heleshawplot(H2)
```

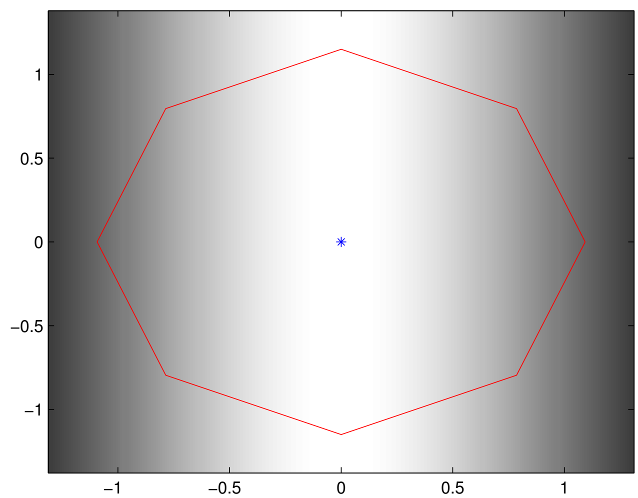

Figure 1: heleshawplot(H1)

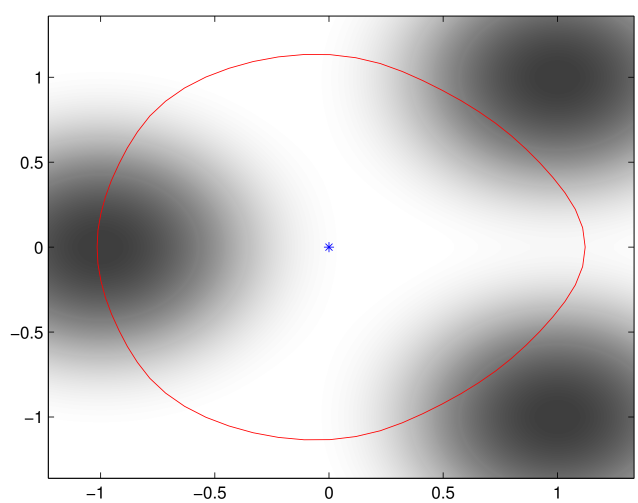

Figure 2: heleshawplot(H2)

4 Related Functions

`bigbox(n)` - return a square polygon of width 2 with `n` vertices in each side

`bigregularpolygon(n)` - return a regular polygon of radius 1 with `n` vertices

`polyedit` - a polygon editor provided by the Schwarz-Christoffel Toolbox

`polyarea` - the area of a polygon. For example if `H` is a heleshaw class then

```
>> p=H.vertices(1:end,n))
>> polyarea(real(p),imag(p))
```

gives the area of the polygon the n-th step

5 Output

The simulation is a discrete form of the Hele-Shaw flow using an approximation of the Polubarinova-Galin equations[1, 1.4.2]. If H_i and H_f denote the initial condition and the final polytope respectively, then the measures

$$A_f = \int_{H_f} f(x+iy)dxdy$$

$$A_i = \int_{H_i} f(x+iy)dxdy$$

should (in an ideal simulation) be related by

$$A_f = A_i + \text{numberofsteps} * \text{timestep}$$

and the higher moments should agree [1, 1.5], that is for all $n \geq 1$ we have

$$\int_{H_i} (x+iy)^n f(x+iy)dxdy = \int_{H_f} (x+iy)^n f(x+iy)dxdy$$

6 Methodology

A single step of the flow is computed as follows. Starting with an initial condition given by a polygon with vertices $p_1, \ldots, p_n$ the Schwarz-Christoffel function $f\colon D \to \mathbb{C}$ is computed using the Schwarz-Christoffel Toolbox, so is a conformal map with $f(0)$ being the point of injection (i.e. the `center`) and takes certain points in the boundary of D to the points p_j.

Using the Polubarinova-Galin equation (modified to take account of a possibly varying permeability encoded by the density function) the infinitesimal change in the points p_j is then approximated using the function f.

Bibliography

[1] B. Gustafsson and A. Vasil'ev *Conformal and potential analysis in Hele-Shaw cells* Advances in Mathematical Fluid Fluid Mechanics. Birkhäuser Verlag, Basel, 2006.

[2] The Schwarz-Christoffel Toolbox http://www.math.udel.edu/ driscoll/SC/