Project Documentation

Aditya Natu

1. Introduction

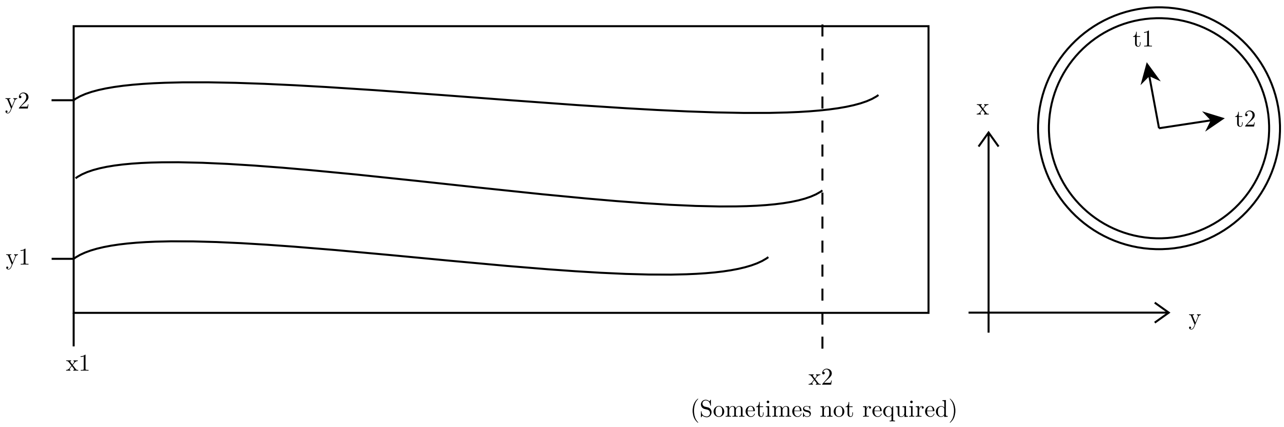
This project includes 3 files for plotting streamlines, streaklines and pathlines. They share a common syntax and work for unsteady as well as steady state problems. It uses Runge-Kutta 4th order method for the required calculations. For unsteady problems, an animation is generated. Minimal use of MATLABs inherent abilities has been made. The program allows the user to choose the number of streamlines, streaklines or pathlines to be plotted. Upon plotting, one can see a vector field in the background, depicting the velocities throughout. If only one streamline is desired, it begins at the center of the leftmost end of the chosen domain. Else, the streamlines are uniformly distributed along the leftmost edge. The same applies to streaklines and pathlines. More details on the working of the program can be found below.

2. Syntax

As said before, the three functions share a common syntax.

```
Streamline(U,V,x1,x2,y1,y2,t1,t2,n)
Streakline(U,V,x1,x2,y1,y2,t1,t2,n)
Pathline(U,V,x1,x2,y1,y2,t1,t2,n)
```

The meaning of the inputs is as below



Parameter	Type	Meaning
U	function	The x-component of the velocity field. U should be a function of three parameters: x,y,t, i.e. the x-coordinate, y-coordinate and time respectively.
V	function	The y-component of the velocity field. U should be a function of three parameters: x,y,t, i.e. the x-coordinate, y-coordinate and time respectively.
x1	real number	The x-coordinate of the leftmost edge of the domain. Nothing to the left of $x = \mathrm{x1}$ will be plotted. Hence, this is an important input.
x2	real number	The x-coordinate of the rightmost edge of the domain. In the function for streamlines, nothing to the right of $x = \mathrm{x2}$ will be plotted. However, in the functions for streaklines, the amount of time chosen for simulation decides the last value of x-coordinate. Hence, for unsteady streaklines and pathlines, the input x2 is *ignored*. For steady streaklines and pathlines, though, x2 is again an important input. After all, steady streaklines and pathlines are the same as steady streamlines.
y1	real number	The lowest point where a curve (stream- streak- or path- line) *begins.* Points below y1 may appear in the curve, though. However, no curve shall start below y1.
y2	real number	The highest point where a curve (stream- streak- or path- line) begins. Points above y2 may appear in the curve, though. However, no curve shall start above y2.
t1	real number	The time at which the plotting must begin. Unsteady streamline animation begins with streamlines at t1. In unsteady streakline and pathline animations, the particles to be traced begin at $t = \mathrm{t1},\ x = \mathrm{x1},\ \mathrm{y1} \leqslant y \leqslant \mathrm{y2}$. If t1 = t2, the functions assume it to be a steady state problem.
t2	real number	The time at which the plotting must end. If t1 = t2, the functions assume it to be a steady state problem.
n	natural number	The number of curves desired. If n=1, one curve, starting at $\left(x = \mathrm{x1},\ y = \dfrac{\mathrm{y1+y2}}{2}\right)$, will be plotted. If n=2, two curves, starting at (x1,y1) and (x1,y2) will be plotted. If n>2, the curves will start on the edge x=x1, uniformly distributed between y=y1 and y=y2.

3. Specifying the velocity fields

The plotting functions are vectorized, and so should be the velocity field functions. The easiest way to do so is to use an anonymus function handle. The velocity functions, whether steady or unsteady, *must* take time as an input. The time input may not have any significance (i.e. may be left unused) in a steady velocity profile.

```
CORRECT WAY:
U = @(x,y,t) (enter your expression here)
V = @(x,y,t) (enter your expression here)

INCORRECT WAY:
U = @(x,y) (enter your expression here)
V = @(x,y) (enter your expression here)
```

For vectorization to be successful, all operations must be specified to be element-wise. See below. Remember, the inputs x,y,t are going to be *matrices.*

```
CORRECT WAY
y.*x
y.^2
y./x

INCORRECT WAY
y*x
y^2
y/x
```

If the velocity fields are steady, one must ensure t1=t2 in the inputs. For steady state, the velocity fields must not depend on time. If they do, t=t1 (=t2) will be substituted and the steady state plots will be generated as usual.

4. Examples

4.1. Unsteady State

Consider a velocity field

$u(x, y, t) = 1 - y^2 + \sin(2\pi y)\ \sin(x)\ \sin(t)$

$v(x, y, t) = \frac{1}{2\pi}(-1 + \cos(2\pi y))\ \cos(x)\ \sin(t)$

to be plotted between $0 < t < 5$. Let the domain be $0 < x < 10$, $0 < y < 1$.
This means t1 = 0, t2 = 5, x1 = 0, x2 = 10, y1 = 0, y2 = 1

Begin by specifying the velocity fields

```
U = @(x,y,t) 1-y.^2 + sin(2*pi*y).*sin(x).*sin(t);
V = @(x,y,t) (1/(2*pi))*(-1+cos(2*pi*y)).*cos(x).*sin(t);
```

Then, call the functions as follows

```
Streamline(U,V,0,10,0,1,0,5,n)
Streakline(U,V,0,10,0,1,0,5,n)
Pathline(U,V,0,10,0,1,0,5,n)
```

This should generate the animations. Refer to the demonstration video to view the output.

4.2. Steady State

Consider a velocity field

$u(x, y, t) = 1 - y^2 + \sin(2\pi y)\ \sin(x)$

$v(x, y, t) = \frac{1}{2\pi}(-1 + \cos(2\pi y))\ \cos(x)$

Let the domain be $0 < x < 10$, $0 < y < 1$.
x1 = 0, x2 = 10, y1 = 0, y2 = 1

Begin by specifying the velocity fields. Observe that U and V are *still written as functions of* t, although the input t remains *unused* in the functions.

```
U = @(x,y,t) 1-y.^2 + sin(2*pi*y).*sin(x);
V = @(x,y,t) (1/(2*pi))*(-1+cos(2*pi*y)).*cos(x);
```

Then, call the functions as follows

```
Streamline(U,V,0,10,0,1,0,0,n) % Ensure that t1=t2. Any value is allowed. For eg, t1=t2=0
Streakline(U,V,0,10,0,1,5,5,n) % Same as above, we took t1=t2=5 this time. Doesn't matter.
Pathline(U,V,0,10,0,1,1,1,n) % Now we took t1=t2=1. Only t1 and t2 being equal is
necessary.
```

This should generate the plots. The plots obtained for streamlines, streaklines, and pathlines will be identical.