WRITEUP

Project part 2

Ines Dormoy
06679317

1 Summary of the overall project

We are considering a system where we are transferring some hot fluid, with temperature T_h, within a pipe. To keep the exterior of the pipe cold, a series of cold air jets, with temperature T_c, are equally distributed along the pipe and continuously impinge on the pipe surface. We are interested in determining the value of the mean temperature within the pipe walls. To model this problem, we will analyze one periodic section of the pipe wall. For this, we discretize the pipe wall and solve equations for each point (see Figure 1).

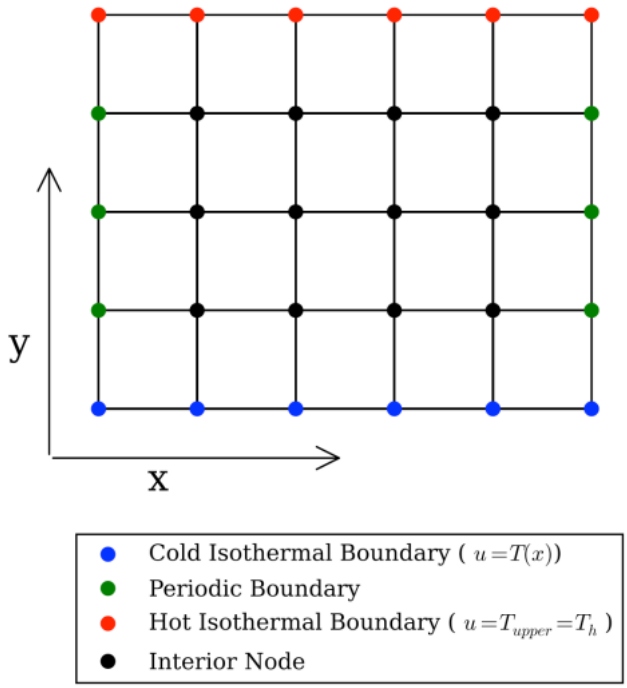

Figure 1: Input1.txt

2 Solver implementation

2.1 Conjugate gradient algorithm

Algorithm 1 Conjugate Gradient Method

initialize u_0
$r_0 = b - Au_0$
L2normr0 = $\|r_0\|$
$p_0 = r_0$
niter = 0
while niter < nitermax **do**
  niter = niter +1
  $\alpha_n = (r_n^T r_n)/(p_n^T A p_n)$
  $u_{n+1} = u_n + \alpha_n p_n$
  $r_{n+1} = r_n - \alpha_n A p_n$
  L2normr = $\|r_{n+1}\|$
  if L2normr/L2normr0 < threshold **then**
    break
  end if
  $\beta_n = (r_{n+1}^T r_{n+1})/(r_n^T r_n)$
  $p_{n+1} = r_{n+1} + \beta_n p_n$
end while

2.2 Matrix and solver classes

Matrix class (SparseMatrix)
Attributes:

- *i_idx* - line indexes
- *j_idx* - columns indexes
- a - values in the matrix
- ncols - number of columns of the matrix;
- nrows - number of rows of the matrix;

Methods:

- AddEntry - add an entry to a matrix
- ConvertToCSR - convert to CSR
- *get_i* - get the vector of lines
- *get_j* - get the vector of columns
- *get_a* - get the vector of values

Solver class (HeatEquation2D)
Attributes:

- A - sparse matrix to be solved
- b - vector in the Ax=b equation
- x - solution vector
- ncols - number of columns of the matrix;
- nrows - number of rows of the matrix;
- ninc - number of unknowns in the system
- niter - number of iterations made by CG algorithm

Methods:

- Setup - sets the system that we need to solve
- Solve - solve the system and return the solution

3 Users guide

Here is the list of commands to execute to get the results:

```
$ make
$ ./main inputXX.txt solution
$ python3 postprocess.py input1.txt solutionXX.txt
```

4 Results

```
$ ./main input0.txt solution
$ SUCCESS: CG solver converged in 8 iterations

$ python3 postprocess.py input0.txt solution10.txt
$ Input file processed: input0.txt
$ Mean Temperature: 100.0002
```

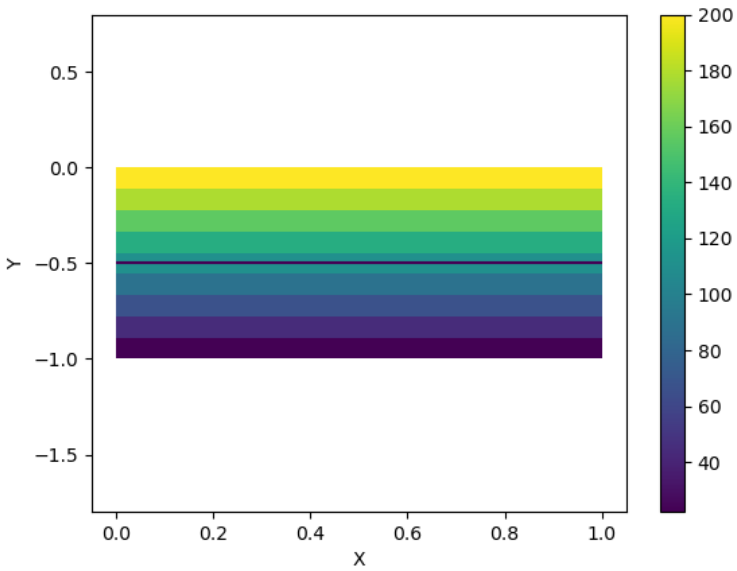

Figure 2: Input0.txt

```
$ ./main input1.txt solution
$ SUCCESS: CG solver converged in 129 iterations
```

```
$ python3 postprocess.py input1.txt solution130.txt
$ Input file processed: input1.txt
$ Mean Temperature: 115.5954
```

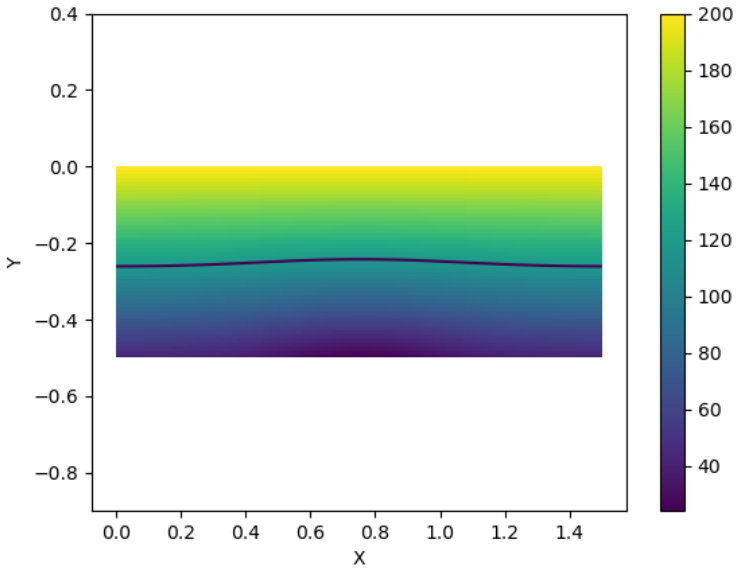

Figure 3: Input1.txt

```
$ ./main input2.txt solution
$ SUCCESS: CG solver converged in 155 iterations

$ python3 postprocess.py input2.txt solution160.txt
$ Input file processed: input2.txt
$ Mean Temperature: 81.0418
```

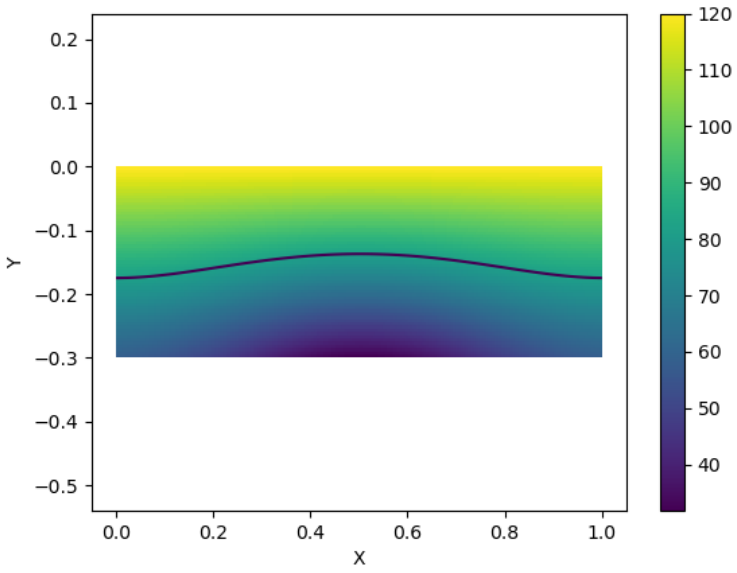

Figure 4: Input2.txt

Please note I didn't ran the python post processing file on rice because of SSH issues (slow connection, no loading of the files). That might explain why the colors were different on my computer for the plots.

5 References

[1] Final Project: Part 1 for CME 211: Software Development for Scientists and Engineers. http://coursework.stanford.edu. Stanford University, Dec 2022.
[2] Final Project: Part 2 for CME 211: Software Development for Scientists and Engineers. http://coursework.stanford.edu. Stanford University, Dec 2022.