Coiled tube reactors under pulsed flow conditions have been shown to demonstrate promising mixing behavior. Optimizing the geometry and operating conditions of coiled tube reactors is important to investigate performance, identify underlying physics and ensure industrial feasibility. Experiments and computational studies have verified that by encouraging radial mixing, and discouraging axial mixing, coiled tube reactors under pulsed flow conditions promote plug-flow performance which enables tighter product distributions [3, 8, 9, 11, 12]. Computational fluid dynamics (CFD) simulations of the system are expensive due to complex vortex phenomena, and gradient information is unavailable. The resulting expensive black-box optimization problem is analogous to hyper-parameter optimization [1, 4, 7, 14]. However, chemical discovery [5, 10], and engineering design [2, 6, 13] are also examples of domains where expensive black-box optimization problems are formulated. For example as follows

$$x^* = \underset{x \in \mathcal{X} \subset \mathbb{R}^d}{\operatorname{argmax}} f(x). \tag{1}$$

In this coursework you will optimise a 2D approximation to a coiled tube reactor system using a machine learning optimisation method of your choice. An example simulation is shown in Figure 1. Make note that

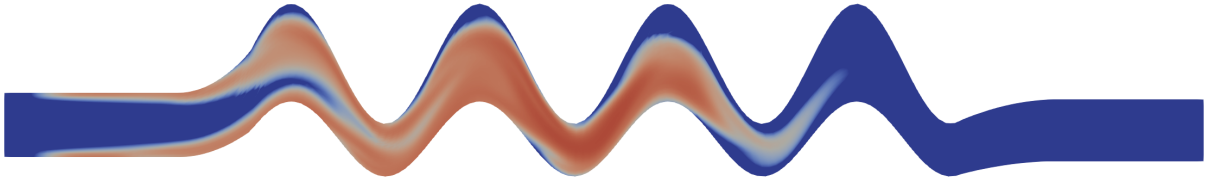


Figure 1: An example simulation.

CFD simulations of the reactor are computationally expensive, and take on the order of 10 minutes. Therefore, you will have to be careful about how many points you sample, and where.

1 System Description

1.1 Parameterisation

The reactor is parameterised by five values: amplitude of pulsed-flow, frequency of pulsed-flow, and three geometry parameters. Examples of reactor geometries and solutions **will not be given**, and we encourage groups to investigate these themselves. This is to encourage the use of machine learning to explore the search space as opposed to attempting to heuristically locate an optimal solution. Table 1 shows the upper and lower bounds for all five parameters.

Table 1: Lower and upper bounds of problem parameters

Parameter	**Lower bound**	**Upper bound**
Amplitude of pulsed-flow	0.001	0.008
Frequency of pulsed-flow	2	8
Geometry parameter 1	0.1	0.7
Geometry parameter 2	3	6
Geometry parameter 3	0	$\pi/2$

1.2 Inside the black-box

The simulation has been set up in such a way that you will only need to provide a set of parameters and the function will return a value which represents the plug-flow performance. **You should not modify this code**. However, in the interest of transparency the following procedures are performed.

1. A folder is created within `outputs/` that is named after the datetime the function is evaluated. This folder will contain all information about a given simulation.

2. Reactor geometry is created from the given parameters, and a preview is saved to the folder, named `reactor_geometry.png`. Subsequently, geometry is then meshed into finite elements.
3. A simulation is then performed in `OpenFOAM` via the `PyFoam` library using the given operating conditions.
4. From the concentration profile that is returned at the outlet of the reactor, a dimensionless number is calculated which represents plug-flow performance.

2 Getting Started

2.1 Instructions

The following process has been set up for you as a way to run CFD simulations on your own computer. Note that whilst this seems long winded: be patient, and read all the steps. This is significantly easier than installing OpenFOAM and all the relevant libraries yourself.

1. Download and install Docker by clicking here and following the instructions.
2. In a terminal, and in the desired location, clone the Git repository. For example in my Documents folder:

```
$ cd Documents
$ git clone https://github.com/OptiMaL-PSE-Lab/ml_coursework.git
```

   If Git is not installed, download and install Git by clicking here and following the instructions.

3. Then enter the Git repository:

```
$ cd ml_coursework
```

4. Build the Docker image:

```
$ docker build --no-cache -t ml_coursework .
```

5. Run the Docker image as a container. This will also enable all the output files to be shared with your host machine.
   - **If you are on Mac or Linux run the following command:**

```
$ docker run -v $(pwd)/:/root/ml_coursework/ -it -p 8888:8888 ml_coursework
```

   - **If you are in the Windows terminal run the following command:**

```
$ docker run -v /cd/:/root/ml_coursework/ -it -p 8888:8888 ml_coursework
```

6. You should now be within the Docker container and should see something ending in `#`. Here enter the following command to start a Jupyter notebook.

```
# jupyter notebook --ip 0.0.0.0 --port 8888 --no-browser --allow-root
```

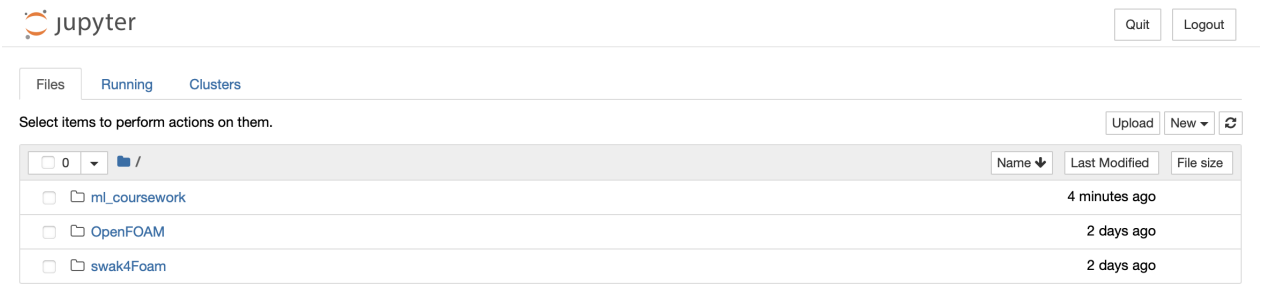
7. The terminal will then prompt you with

```
    To access the notebook, open this file in a browser:
      file:///root/.local/share/jupyter/runtime/nbserver-640-open.html
Or copy and paste one of these URLs:
     http://41b527251408:8888/?token=044321e3fdb7493b611b493d...
 or http://127.0.0.1:8888/?token=044321e3fdb7493b611b493d80d23...
```

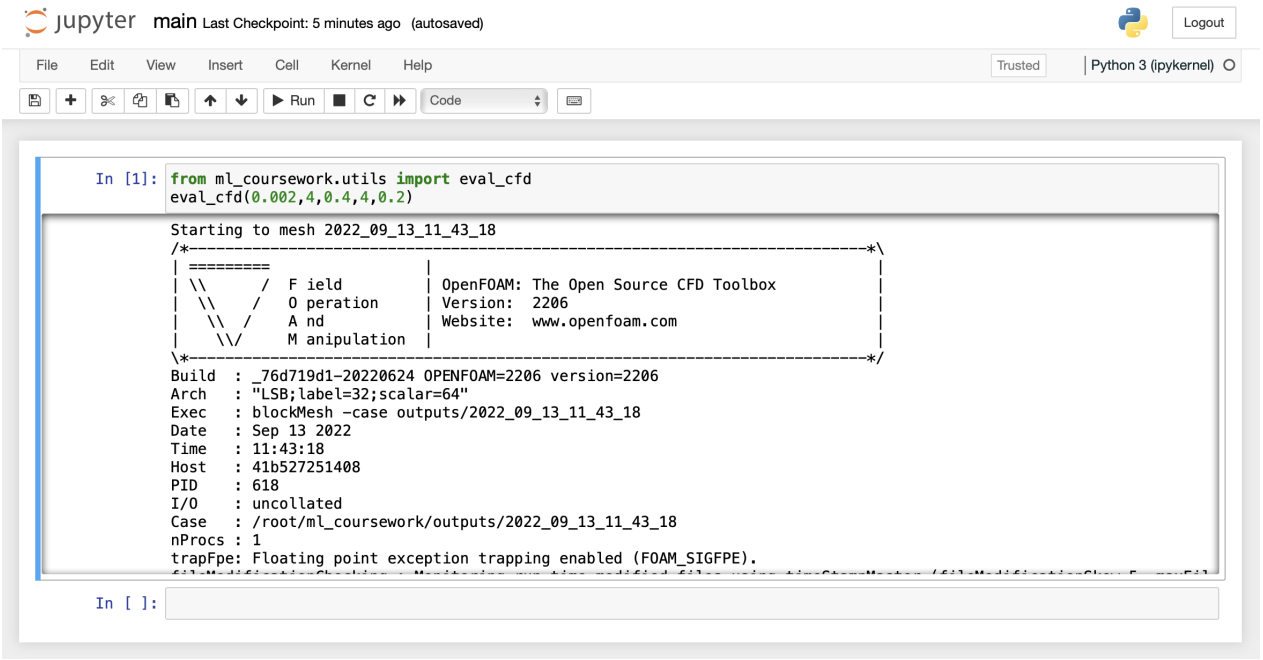
   I prefer to use the last URL which I copy and paste into my browser. Note these URLs are just examples that I have generated and they will not work for your machine.

2.2 Jupyter Interface

In your browser you should now see the following



If you enter the `ml_coursework` folder you will find everything you need, mainly the `main.ipynb` file. If you click on this file you will find the single function which evaluates the CFD code. If you run the cell, the output should look something like this:



To check the simulation is running properly, check in the folder where you cloned the Git repository. The a folder containing this simulation

References

[1] Bergstra, J., Bardenet, R., Bengio, Y., and Kégl, B. Algorithms for Hyper-Parameter Optimization. In *Advances in Neural Information Processing Systems* (2011), J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Weinberger, Eds., vol. 24, Curran Associates, Inc.

[2] Daniels, S. J., Rahat, A. A. M., Tabor, G. R., Fieldsend, J. E., and Everson, R. M. Automated shape optimisation of a plane asymmetric diffuser using combined Computational Fluid Dynamic simulations and multi-objective Bayesian methodology. *International Journal of Computational Fluid Dynamics 33*, 6-7 (Aug. 2019), 256–271.

[3] de Brito Lira, J. O., Riella, H. G., Padoin, N., and Soares, C. Fluid dynamics and mass transfer in curved reactors: A CFD study on Dean flow effects. *Journal of Environmental Chemical Engineering 10*, 5 (Oct. 2022), 108304.

[4] Feurer, M., and Hutter, F. Hyperparameter Optimization. In *Automated Machine Learning*. Springer International Publishing, 2019, pp. 3–33.

[5] JORAYEV, P., RUSSO, D., TIBBETTS, J. D., SCHWEIDTMANN, A. M., DEUTSCH, P., BULL, S. D., AND LAPKIN, A. A. Multi-objective Bayesian optimisation of a two-step synthesis of p-cymene from crude sulphate turpentine. *Chemical Engineering Science 247* (Jan. 2022), 116938.

[6] LAM, R., POLOCZEK, M., FRAZIER, P., AND WILLCOX, K. E. Advances in Bayesian Optimization with Applications in Aerospace Engineering. In *2018 AIAA Non-Deterministic Approaches Conference* (Jan. 2018), American Institute of Aeronautics and Astronautics.

[7] LARSON, J., MENICKELLY, M., AND WILD, S. M. Derivative-free optimization methods. *Acta Numerica 28* (2019), 287–404.

[8] MANSOUR, M., KHOT, P., THÉVENIN, D., NIGAM, K. D., AND ZÄHRINGER, K. Optimal Reynolds number for liquid-liquid mixing in helical pipes. *Chemical Engineering Science 214* (Mar. 2020), 114522.

[9] MANSOUR, M., LIU, Z., JANIGA, G., NIGAM, K., SUNDMACHER, K., THÉVENIN, D., AND ZÄHRINGER, K. Numerical study of liquid-liquid mixing in helical pipes. *Chemical Engineering Science 172* (Nov. 2017), 250–261.

[10] MATEOS, C., MARÍA JOSÉ NIEVES-REMACHA, AND RINCÓN, J. A. Automated platforms for reaction self-optimization in flow. *Reaction Chemistry &amp Engineering 4*, 9 (2019), 1536–1544.

[11] MCDONOUGH, J., AHMED, S., PHAN, A., AND HARVEY, A. The development of helical vortex pairs in oscillatory flows – A numerical and experimental study. *Chemical Engineering and Processing - Process Intensification 143* (Sept. 2019), 107588.

[12] MCDONOUGH, J., MURTA, S., LAW, R., AND HARVEY, A. Oscillatory fluid motion unlocks plug flow operation in helical tube reactors at lower Reynolds numbers (Re$\leq$10). *Chemical Engineering Journal 358* (Feb. 2019), 643–657.

[13] MORITA, Y., REZAEIRAVESH, S., TABATABAEI, N., VINUESA, R., FUKAGATA, K., AND SCHLATTER, P. Applying Bayesian optimization with Gaussian process regression to computational fluid dynamics problems. *Journal of Computational Physics 449* (Jan. 2022), 110788.

[14] YU, T., AND ZHU, H. Hyper-parameter optimization: A review of algorithms and applications, 2020.