Thermal Storage Design: Part 1

ETH zürich

Deep Learning in
Scientific Computing
Due date: June 12th, 2022

Training and testing data can be found on the Moodle page: `https://moodle-app2.let.ethz.ch/course/view.php?id=17291`

IMPORTANT INFORMATION

To get ECTS credits for the course Deep Learning in Scientific Computing you need to submit and obtain a passing grade on the project. The project consists of two parts and the **submission deadline is the 12th of June** for both of them. Each part accounts for several tasks where you will be asked to train a learning model and provide predictions on a testing set. Therefore, **the final submission consists of the predictions files together with a report of maximum 2000 characters** for each task where you succinctly describe the procedure followed. Those have to be collected in a zip folder named as *yourfirstname_yoursecondname_yourleginumber.zip*. **Only the students that submit the report and the prediction files will be graded.**

Project Description

The main objective of the project is to apply machine learning algorithms to solve various tasks related to the preliminary design of a *thermal energy storage*.

The device is used in solar power plants to store thermal energy during the *charging phase* and release it for production of electricity during the *discharging phase*. The thermal energy is stored due to the interaction of a fluid and a solid phase. During the charging state the fluid is injected at high temperature from one end of the storage and heats the solid up. In contrast, during the discharging phase the reverse process occurs: cold fluid flows from the opposite end and absorbs heat from the solid. Between charging and discharging *idle phases* take place, where no fluid enters the thermal storage.

Therefore, at any instant of time the thermal storage can be in one of the following states:

1. Charging;
2. Idle between charging and discharging;
3. Discharging;
4. Idle between discharging and charging;

Together the four states establish a *cycle* and the same process is repeated for several cycles until the thermal storage reaches a periodic or stationary regime.

Mathematical Model

The thermal storage is modeled by a cylinder with length L and diameter D and it is assumed that temperature variation occurs only along the axis of the cylinder.

The temperature evolution of the solid and fluid phases, T_s and T_f, is described by a system of two linear *reaction-convection-diffusion* equations:

$$\begin{aligned} \varepsilon\rho_f C_f \frac{\partial T_f}{\partial t} + \varepsilon\rho_f C_f u_f \frac{\partial T_f}{\partial x} &= \lambda_f \frac{\partial^2 T_f}{\partial x^2} - h_v(T_f - T_s) \quad x \in [0, L],\ t \in [0, T], \\ (1-\varepsilon)\rho_s C_s \frac{\partial T_s}{\partial t} &= \lambda_s \frac{\partial^2 T_s}{\partial x^2} + h_v(T_f - T_s) \quad x \in [0, L],\ t \in [0, T], \end{aligned} \tag{1}$$

with ρ being the density of the phases, C the specific heat, λ the diffusivity, ε the solid porosity, u_f the fluid velocity entering the thermal storage and h_v the heat exchange coefficient between solid and fluid. The fluid velocity is assumed to be uniform along the cylinder and varying only in time: $u_f = u$ during charging, $u = 0$ during idle and $u_f = -u$ during discharging, with u being a positive constant.

The system of equations has to be augmented with suitable initial and boundary conditions:

$$T_f(x, t = 0) = T_s(x, t = 0) = T_0,\ x \in [0, L] \tag{2}$$

$$\frac{\partial T_s(x,t)}{\partial x}\Big|_{x=0} = \frac{\partial T_s(x,t)}{\partial x}\Big|_{x=L} = 0,\ t \in [0, T] \tag{3}$$

The boundary conditions for the fluid instead will be different according to the current state of the thermal storage:

- **Charging State:**

$$T_f(0,t) = T_{hot}, \qquad \frac{\partial T_f(x,t)}{\partial x}\Big|_{x=L} = 0, \qquad t \in [0,T] \tag{4}$$

- **Discharging State:**

$$\frac{\partial T_f(x,t)}{\partial x}\Big|_{x=0} = 0, \qquad T_f(L,t) = T_{cold}, \qquad t \in [0,T] \tag{5}$$

- **Idle Phase:**

$$\frac{\partial T_f(x,t)}{\partial x}\Big|_{x=0} = 0, \qquad \frac{\partial T_f(x,t)}{\partial x}\Big|_{x=L} = 0, \qquad t \in [0,T] \tag{6}$$

Task 1: Noisy Function Approximation

Let us assume that **noisy** measurements of the fluid and solid temperature $T^0_{s,i}, T^0_{f,i}$ are taken at the top end of the storage $x = 0$ during the entire process.

The objective of the first task is to approximate the maps

$$T^0_s : t \mapsto T_s(0,t) \quad T^0_f : t \mapsto T_f(0,t)$$

with suitable learning models $T^{0,*}_s \approx T^0_s$ and $T^{0,*}_f \approx T^0_f$ given the measurements $(t_i, T^0_{s,i}, T^0_{f,i})$, $i = 1, ..., N$. The training data can be found in the file **Task1/TrainingData.txt** (see figure 1 for a representation of the training data).

Once the model is trained, use it to make predictions on the test set that can be found in the file **Task1/TestingData.txt** and save your predictions in the folder *yourfirstname_yoursecondname_yourleginumber* as *Task1.txt*. **The format of the file has to be the same as the file Task1/SubExample.txt.** Observe that the $i-th$ row of the submission file has to correspond to the $i-th$ row of the testing set. Therefore, make sure not to change the order of the testing samples.

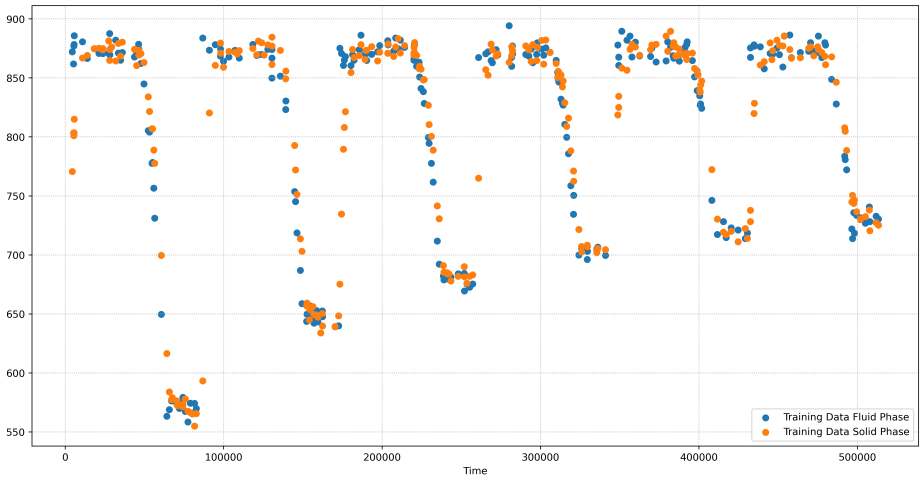

Figure 1: Task 1 training data

Task 2: Learning Observables in High Dimensional Space

In order to evaluate the performance of the storage, non-dimensional *observables* are usually defined. One of them is the capacity factor CF:

$$CF = \frac{Q_{b.d} - Q_{b.c}}{Q_{max}}$$

with $Q_{b.d} = Q(t_{b.d})$ and $Q_{b.c} = Q(t_{b.c})$ being the thermal energy Q stored before discharging and charging, and Q_{max} the maximal thermal energy that can be potentially stored:

$$Q(t) = \frac{\pi}{4} D^2 \left[\varepsilon \rho_f C_f \int (T_f - T_{cold}) dx + (1-\varepsilon) \rho_s C_s \int (T_s - T_{cold}) dx \right], \tag{7}$$

$$Q_{max} = \frac{\pi}{4} D^2 L (T_{hot} - T_{cold}) [\varepsilon \rho_f C_f + (1-\varepsilon) \rho_s C_s]. \tag{8}$$

In this task we want to learn the map from a set of several input variables to the capacity factor. The set of input variables includes: the density of the solid and fluid phase (ρ_s, ρ_f), the specific heats (C_s, C_f), the massflow of fluid phase entering the storage (m_f), the diameter of the elements of the solid phase (d), and the volume and diameter of the storage (V, D).

In particular, let us consider the following parametrization of the input variables:

$$\begin{aligned} \rho_s &= \rho_s^0 (1 + \sigma G_1(y^{(1)})), & \rho_f &= \rho_f^0 (1 + \sigma G_2(y^{(2)})) \\ C_s &= C_s^0 (1 + \sigma G_3(y^{(3)})), & C_f &= C_f^0 (1 + \sigma G_4(y^{(4)})) \\ m_f &= m_f^0 (1 + \sigma G_5(y^{(5)})), & d &= d^0 (1 + \sigma G_6(y^{(6)})) \\ D &= D^0 (1 + \sigma G_7(y^{(7)})), & V &= V^0 (1 + \sigma G_8(y^{(8)})) \end{aligned} \tag{9}$$

where $y^{(j)} \in [0,1]$ and $G_j(y^{(j)}) = 2y^{(j)} - 1$ for $j = 1, ..., 8$. In this task you are asked to approximate the map:

$$CF : y \mapsto CF(y)$$

with a a suitable learning model CF^*,

$$CF^* \approx CF : y \mapsto CF(y),$$

based on a training set S. Here $y \in [0,1]^8$ denotes the collection of all the input variables $y^{(j)}$,

$$y = [y^{(1)}, y^{(2)}, y^{(3)}, y^{(4)}, y^{(5)}, y^{(6)}, y^{(7)}, y^{(8)}]$$

In order to generate the training set we consider a sequence of low-discrepancy Sobol points $S = \{y_k\}_{k=1}^{160}$,

$$y_k = [y_k^{(1)}, y_k^{(2)}, y_k^{(3)}, y_k^{(4)}, y_k^{(5)}, y_k^{(6)}, y_k^{(7)}, y_k^{(8)}] \in [0,1]^8,$$

and for each realization y_k:

1. transform y_k according to (9);
2. solve the system of equations (1) on a mesh with $N^{(4)} = 1601$ spatial points for each realization y_k of the input parameters;
3. compute the corresponding value of the capacity factor CF_k.

The generated data are collected in the file **Task2/TrainingData_1601.txt**. The first 8 columns of the file corresponds to the **transformed** input variables, whereas the last column contains the corresponding values of the capacity factor.

You are also provided with the additional data files **Task2/TrainingData_401.txt** and **Task2/TrainingData_101.txt** containing the training sets $S^{(\ell)}$, $\ell = 2, 0$ generated as described before at different mesh resolution $N^{(\ell)} = 401, 101$. The training sets $S^{(\ell)}$ account for a total number of 640 and 2560 samples. The original sequence of Sobol point $\{y_k\}_{k=1}^{2560}$ is contained in the file **Task2/samples_sobol.txt**.

Once the approximate map is obtained, use the trained model to make predictions on the test set **Task2/TestingData.txt** and save your predictions in a file called *yourfirstname_yoursecondname_yourleginumber/Task2.txt.* **The format of the file has to be the same as the file Task2/SubExample.txt.** Observe that the $i-th$ row of the submission file has to correspond to the $i-th$ row of the testing set. Therefore, make sure not to change the order of the testing samples.

Task 3: Time Series Forecasting

Consider again the time series described in Task 1. Similarly to the first task we want to approximate the maps:

$$T_f^0 : t \mapsto T_f(0,t), \quad T_s^0 : t \mapsto T_s(0,t)$$

However, in this task we are interested in **future** (or **out of samples**) predictions of the fluid and solid temperature. In other words, the training data consists of time measurements t_i registered in a frame $[0, T]$ and we want to forecast the fluid and solid temperature in the frame $[T, T_{end}]$.

Train a suitable learning model to solve the task based on the data given in **Task3/TrainingData.txt** and make predictions on the test set **Task3/TestingData.txt**. Afterwards, save your predictions in the file *yourfirstname_yoursecondname_yourleginumber/Task3.txt*. **The format of the file has to be the same as the file Task3/SubExample.txt.**

Remark. *The task can be successfully solved by using simple feed-forward dense neural networks. However, you are free to explore other network architectures, conceived specifically to solve this type of tasks, for instance recurrent neural networks (RNN), Long Short-term Memory (LSTM), etc.*