



ECN  
MSc COMPUTATIONAL MECHANICS  
Sem 3

---

## Model Reduction

---

**Lab 1**

Prasad ADHAV

## 1 Exercise 1

The given code and the variables as mentioned in Lab 1 were studied and understood. Then for the first exercise, the `script.m` is compiled. It was observed that after commenting out mPOD part, rest of the code compiled perfectly. Although no results were plotted. After including the surf plot i.e `surf(X, T, uFEM)` and other plots, the complete FEM solution surf image and its cut sections are obtained as shown in figure 1. This solution is obtained for default values available in code.

As defined in the problem statement, at initial time, the temperature is constant ( $T = 0$ ) over the whole bar, and there is no flux going in or out. The material properties defined are (assuming units are consistent), thermal conductivity  $k = 1$ , density  $\rho = 1$ , and the flux  $q = 1$  and specific heat  $c_p = 1$ . According to the Heat transfer equation,  $q = -kA\frac{dT}{dx}$ , substituting all the values, we know that the temperature difference of increase in temperature due the given heat flux should be 1unit.

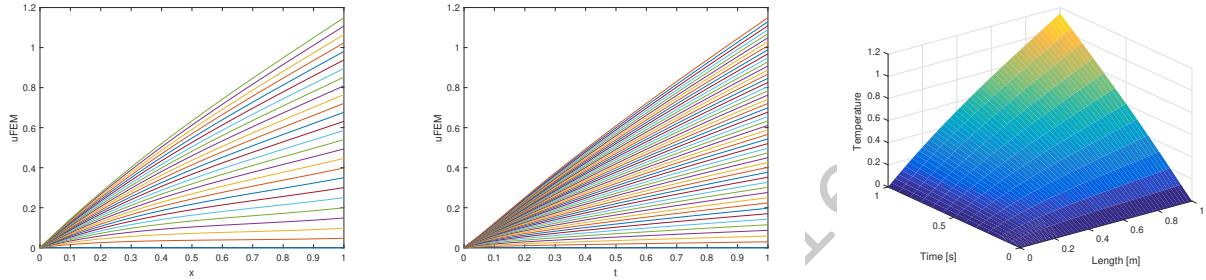


Figure 1: Default FEM solution

As seen in fig 1, the temperature value is 1.14unit. And since the governing equation was a transient equation taking in to account for some more parameters, like  $c_p$ ,  $\rho$ , and FEM was used to solve the equation, a 14% error still looks high. Just to confirm if the underlying physics works properly, the parameters thermal conductivity, density or specific heat can be varied. Since,  $k$ ,  $\rho$  and  $c_p$  are assumed linearly related to each other, it is sufficient to vary only one of the three parameters. We chose to vary  $c_p$  initially, since in the next exercises we are varying  $c_p$ . As observed in fig 2 Since the specific heat of the bar is high (2 (a)), its temperature increases slowly, and increases till only 0.15units. But when we decrease the specific heat, it heats up rapidly to 1.5units. Hence, we can say that the underlying physics is preserved. Also, the time is set to 40sec, and same simulation is done with the same  $c_p$  values as mentioned above.

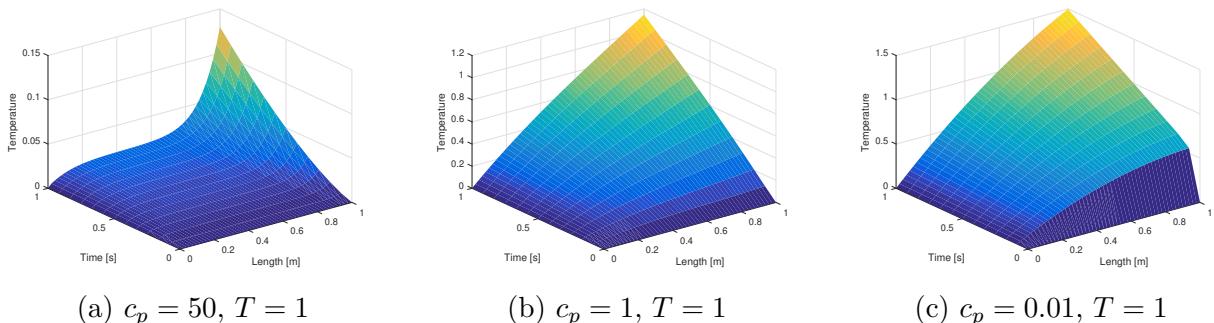


Figure 2: Effect of varying  $c_p$  on the default solution obtained

It can be clearly observed in 2(c), that due to very low  $c_p$ , heating occur very fast, and since the time available is just 1sec, we can see a jump(discontinuity) in the solution obtained. But when we use a longer time interval, it is clear that when sufficient time is allowed for heating we obtained a continuous solution, as shown in fig 3.

From the above figures we can conclude that the physics is represented correctly. And we inferred that we have a satisfactory understanding of the provided code.

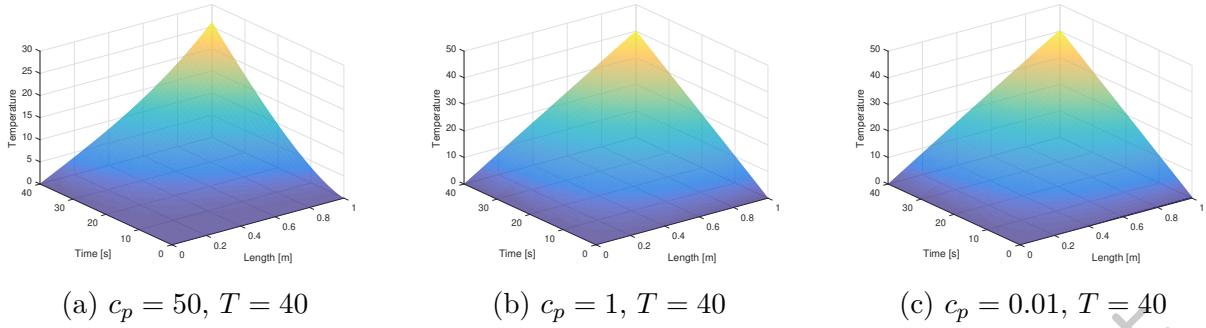


Figure 3: Effect of varying  $c_p$  on the solution for simulation time of 40 sec

## 2 Exercise 2

In exercise 2 the heating process is simulated. The flux is defined as shown in the fig 4. The material properties are set to  $k = 1$ ,  $\rho = 1$  and  $c_p = 50$ . The time interval of interest is 40sec, using time increments of 0.08sec. The full simulation was performed, and the results were saved as Full\_Solution.

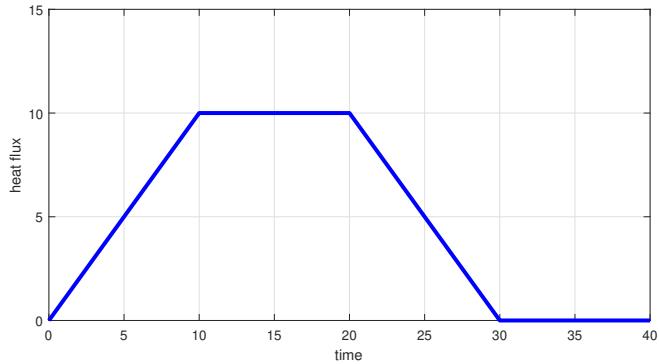


Figure 4: Flux v. Time

After applying all the specified material properties and time constraints, along with the boundary conditions, the surface plot in fig 5 (a) was obtained. The Dirichlet boundary condition says that the temperature at the left end of the bar remains constant( $= 0$ ). Simultaneously, a Neumann boundary conditions is applied which says that the flux at the left end of the bar is zero.

This imposition of BC can be observed in fig 5 (a), for all time steps. But if we consider the physics. The problem statement also states that there is a flux coming in from the right end, as shown in fig 4, and then there is no flux for last 10sec of the simulation. According to physics, for last 10sec the flux is zero at both ends, i.e heat can't come in or go out. Due to this the heat gained by the bar should be theoretically distributed along the whole bar, and the bar should have constant temperature over its length. But due to imposition of Neumann as well as Dirichlet BC, we can see that the physics is violated and the left end temperature remains constants( $= 0$ ). Also, mathematically speaking it is incorrect to impose both Neumann and Dirichlet boundary conditions simultaneously.

This problem was solved by redefining the Dirichlet boundary condition on the left end of the bar. There is enough time available for the bar to attain a constant temperature over its length and this can be seen in fig 5 (b). And now we can say that this simulation is more close to the real 1D conduction problem.

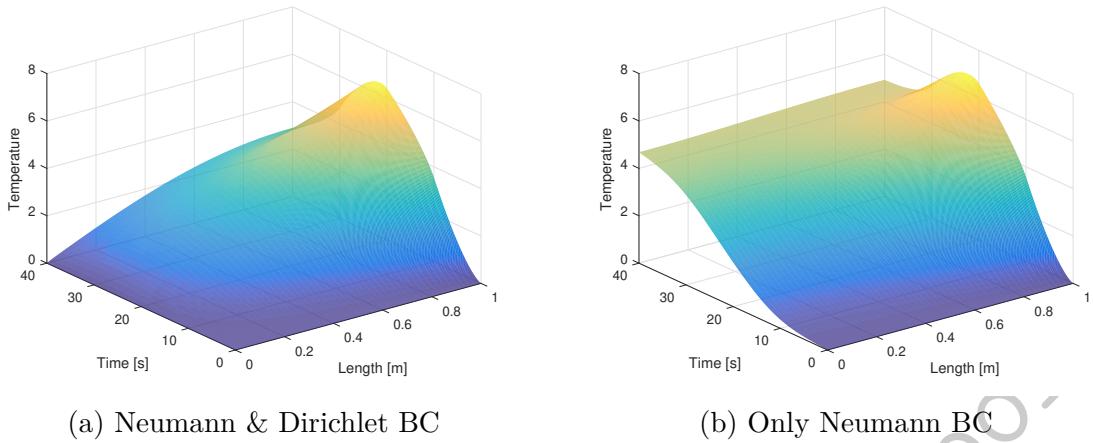


Figure 5: Full Solution

### 3 Exercise 3

As all the parameters stay constant for the 5 seconds analysis, we only took the values of the full computation representing the first 5 seconds of the simulation. We can see the temperature distribution in detail in figure 6. The sample has as boundary conditions a continuously increasing flux in the right end of the bar, causing an increasing heat distribution.

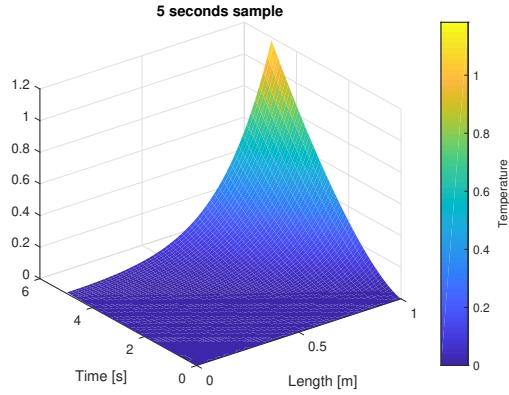


Figure 6: 5 first seconds analysis with  $\epsilon = 10^{-8}$

### 4 Exercise 4

The POD model was obtained after completing the pod.m function, using 50 samples. The computation time is incredibly reduced from the full model to the reduced one, although the scale of such time is so low that the values vary from one test to the other, all parameters constant (most probably due to unrelated internal CPU processes of the same scale as the computational time). Although 50 snapshots may seem an excessive number of samples (and it would make sense, for it represents the 79% of the 63 available samples), the low cost of the problem allowed us to consider that many number of examples. If we take a look at figure 7 we can see that the vast majority of the eigenvalues have an extremely low order of magnitude, with a ratio of  $10^{16}$  between the highest and the last 30 eigenvalues (the flat line in figure 7). Although the first 4 eigenvalues show a relatively high weight, they do not reproduce well enough the solution, needing lower weight eigenvalues (and their associated eigenvectors) to be taken into account for the reduced order basis.

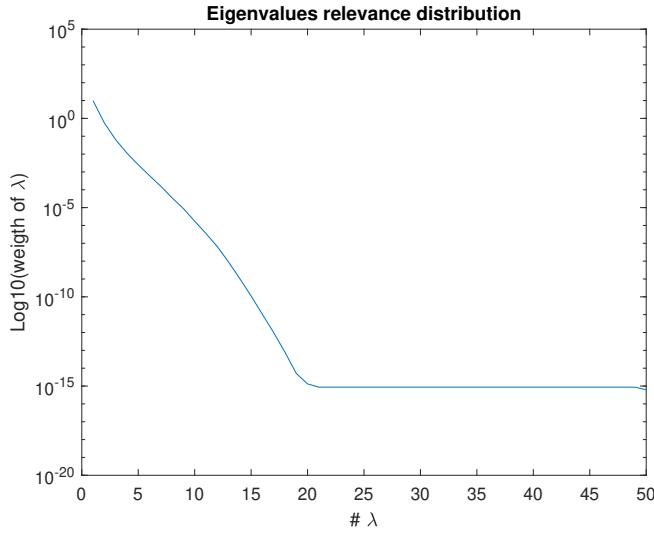


Figure 7: Eigenvalues magnitude distribution obtained after considering 50 samples

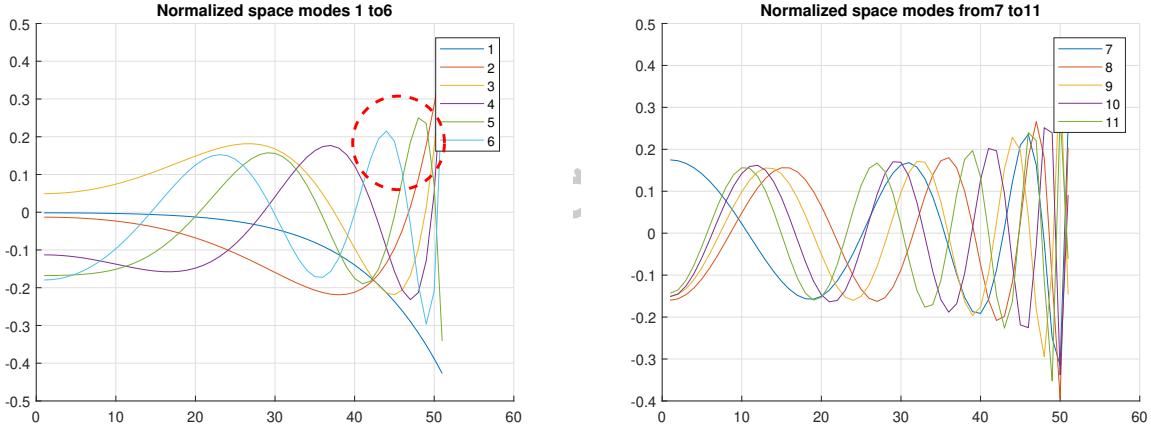


Figure 8: First 11 reduced basis vectors (normalized space modes). The red circle in first plot marks the region where the first 4 modes fail to represent the solution

The first 11 basis vectors can be seen in figure 8, and they could be understood as the main modes on which the solution depends. As we take more modes into our basis, they become more similar (in magnitude and frequency), so that they add less information into the system (something that can also be assessed by looking at their associated eigenvalue). But there is also a minimum number of vectors to include in the basis. In figure 8 we can see how the first four vectors fail at representing the whole solution if the red marked region. We'll have to find a threshold  $\epsilon$  that builds a system of at least size 6, not less.

## 5 Exercise 5

After exploring a range of threshold values we came to the conclusion that a satisfactory reduced order solution was obtained by setting the threshold at  $\epsilon = 10^{-8}$ . The maximum absolute error  $e = |u_{FEM} - POD|$  was of the order of  $10^{-3}$  degrees, which is around  $10^{-2}$  of relative error (with respect to the full order solution  $u_{FEM}$ ). We can see the plots of the absolute and logarithmic relative error  $\log_{10}(\frac{|u_{FEM} - POD|}{u_{FEM}})$  in figure 9. While for most of our domain of interest the relative error values are similar, we can see two regions of special interest:

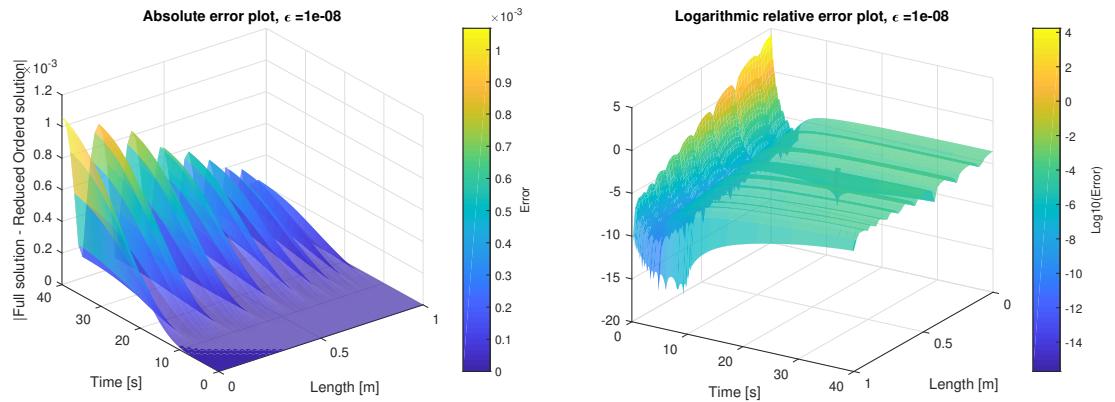


Figure 9: Error plots for the reduced model built with  $\epsilon = 10^{-8}$

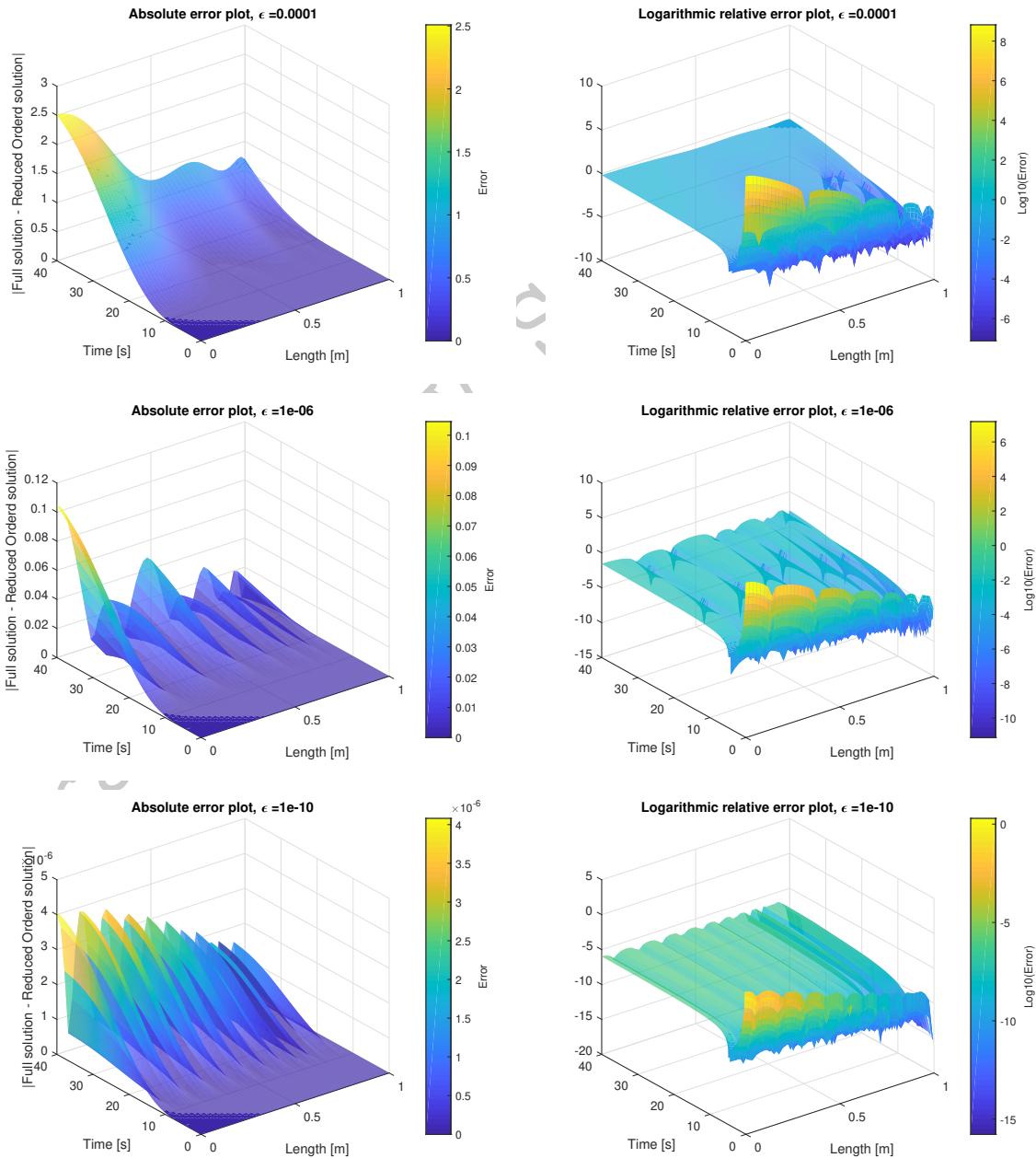


Figure 10: Absolute and relative error plots for  $\epsilon = 10^{-4}$ ,  $\epsilon = 10^{-6}$  and  $\epsilon = 10^{-10}$

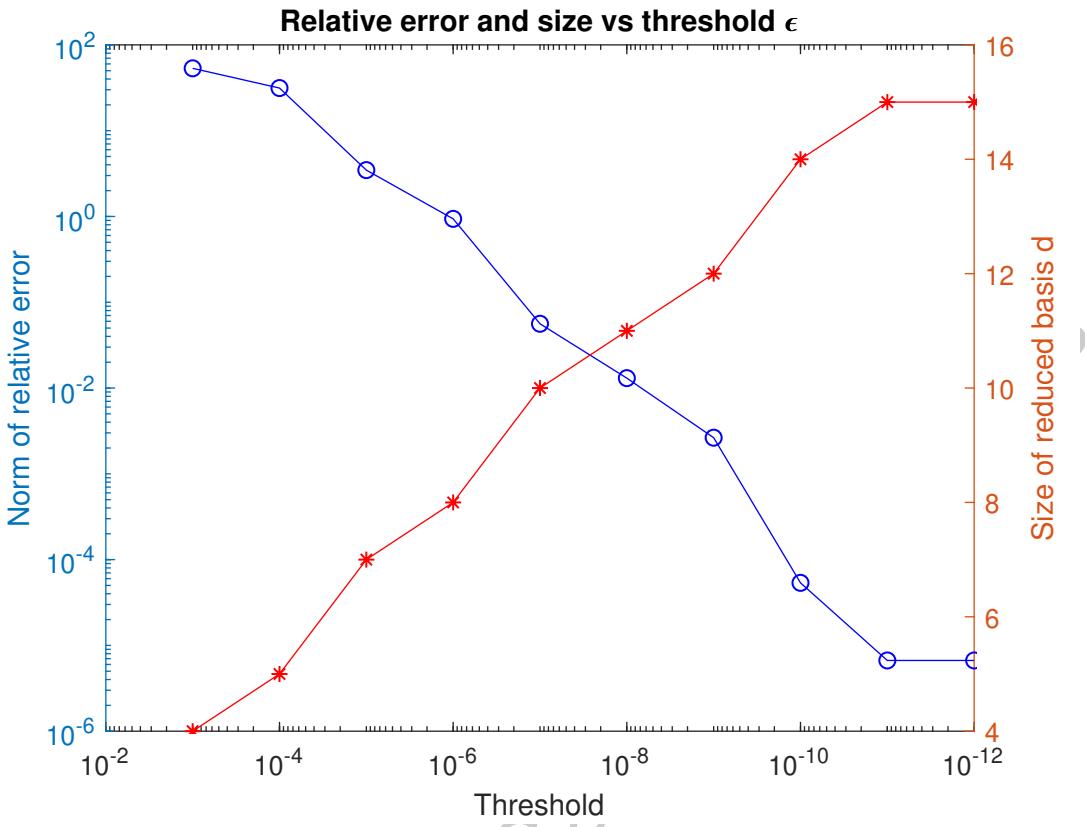


Figure 11: Evolution of the reduced model solution and basis size with respect to the threshold value

1. Points where temperature is close to 0 degrees, corresponding to the first seconds of the simulation
2. The 5 seconds sampling region, excluding the previously commented points

About the first region, the model obtains relative errors of the order of  $10^4$ , while showing negligible absolute errors, as the values are close to zero (for instance,  $10^{-14}$  degrees in the full order solution compared to  $10^{-10}$  degrees in the reduced order solution). The second region shows the best estimation of the solution, having a relative error of (approximately) between  $10^{-6}$  and  $10^{-10}$ . As the model is built using the data extracted from these points, there is no surprise in them having the best predicted temperature. Figure 10 shows the different absolute and logarithmic relative error plots for several values of threshold  $\epsilon$ , exactly as they were shown in figure 9 for  $\epsilon = 10^{-8}$ . We can see that for every time point and threshold considered the highest errors are located at the left end of bar. This is easily seen in the absolute plots but not so much in the relative error plots. In figure 11 we can take a look at the evolution of the 2-norm of the relative error ( $\|error\|_2 = \sqrt{\lambda_{max}^{error}}$ ) and of the size of the basis with respect to the threshold value. We can see that the accuracy increases rapidly at first, and then settles as we include more and more terms. For those basis built with  $\epsilon > 10^{-6}$  we have a relative error over 1, meaning that they are poor approximations. Our chosen  $10^{-8}$  has an accuracy of the order of  $10^{-2}$  (accurate up to two decimal digits). If we are not dealing with extreme precise applications, this may be well enough to model the heating of the bar (although the problem is so simple that there would not be a real need to compute the reduced order model).