Doctoral Thesis

MÜLLER Michael

michaelm@fel.zcu.cz

2023-07-12

All rights reserved©

Errors and omissions excepted

Suggestions and discussions welcome, just leave a message

References

[Ben+21] Peter Benner et al. *Model Order Reduction: Basic Concepts and Notation*. De Gruyter, 2021.

[BM+21] Nanda Kishore Bellam Muralidhar et al. "Parametric model order reduction of guided ultrasonic wave propagation in fiber metal laminates with damage". In: *Modelling* 2.4 (2021), pp. 591–608.

[Che99] Yong Chen. "Model order reduction for nonlinear systems". PhD thesis. Massachusetts Institute of Technology, 1999.

[Got97] Irving Gottlieb. *Practical electric motor handbook*. Elsevier, 1997.

[Kas+19] Jan Kaska et al. "Optimalizace rotoru reluktančního motoru". In: (2019).

[Rib+21] Ângela M Ribau et al. "Flow Structures Identification through Proper Orthogonal Decomposition: The Flow around Two Distinct Cylinders". In: *Fluids* 6.11 (2021), p. 384.

[Sah17] SK Sahdev. *Electrical machines*. Cambridge University Press, 2017.

[Sch08] Wil Schilders. "Introduction to model order reduction". In: *Model order reduction: theory, research aspects and applications* (2008), pp. 3–32.

[Wil06] Theodore Wildi. *Electrical machines, drives, and power systems*. Pearson Educacion, 2006.

alert
This indicates an alert, passage is either wrong, confusing, misleading, or any other kind of high attention.

revise
This indicates a revision, in general it's not wrong but explanation is poor, wording is improper, or not satisfying in general.

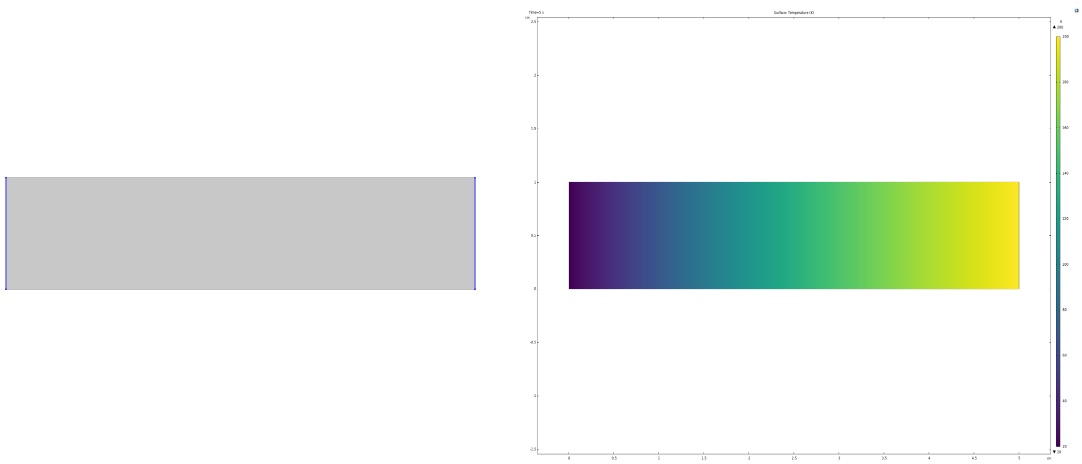
info
This is an information, it contains some side information/comments that can be useful.

under construction
This box contains information/ideas that are not yet formed into sentences, in other words this will be implemented next

1 Setup and results

1.1 I-shape

Before diving into a real and complex setup I decided it is best to start with basic examples where calculations can be done by hand and prototyping yields to faster results. For this the most simplest setup was taken, a simple 2d rectangular setup with fixed temperature $T_0 = 20\,\mathrm{K}$ on the left and $T_1 = 200\,\mathrm{K}$ on the right. The ambient temperature T_{amb} was given by $T_{\mathrm{amb}} = 293.15\,\mathrm{K}$. The geometry is modelled as copper and only the thermal conductivity κ was used as material property. This property and value, however, does not play a major role in the benchmark setup. Same goes with mesh quality. For quick prototyping reasons coarse meshes were accepted.



(a) Geometry setup with fixed temperature on the left T_0 and right T_1 (highlighted in blue).

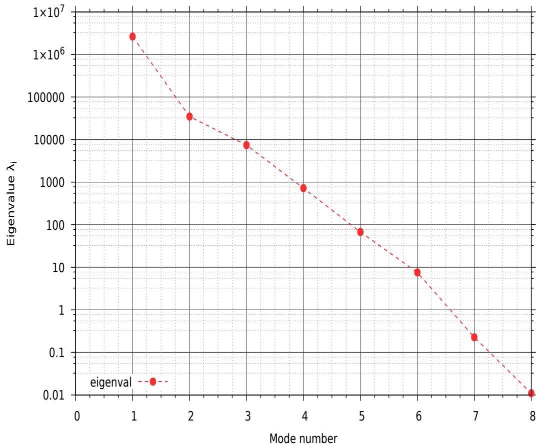
(b) Temperature distribution after $5\,\mathrm{s}$.

Fig. 1.1

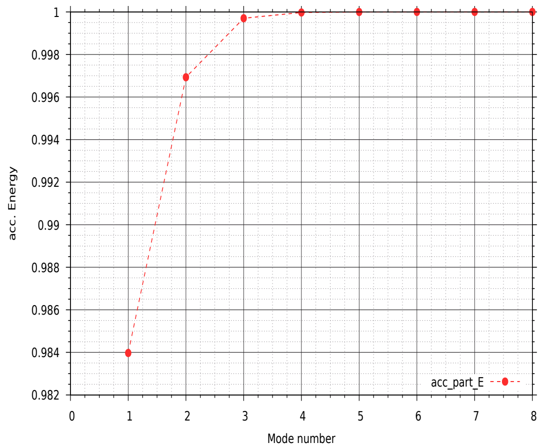
POD	eigenval	acc eigenval	part E	acc part E
1	2.641702e+06	2.642e+06	9.840e-01	0.983972
2	3.478282e+04	2.676e+06	1.296e-02	0.996928
3	7.451407e+03	2.684e+06	2.775e-03	0.999703
4	7.218385e+02	2.685e+06	2.689e-04	0.999972
5	6.723362e+01	2.685e+06	2.504e-05	0.999997
6	7.572878e+00	2.685e+06	2.821e-06	1.000000
7	2.260862e-01	2.685e+06	8.421e-08	1.000000
8	1.093760e-02	2.685e+06	4.074e-09	1.000000

Tab. 1.1: First 8 modes are listed. *eigenval* column listed the eigenvalues in descending order (ordered from most to least important so to speak), *acc eigenval* the sum of all eigenvalues up to this row, *part E* columns represents the percentual energy each eigenvalue contributes to the system and the last column *acc part E* the accummulated partial energy up to this row.

In this setup the energy stored is $E_{\mathrm{stored}} = 2.685 \times 10^6$ and the first mode is contributing more than $98\,\%$ of this value. That means that the first mode dominates the system. Having a look on the following left graph demonstrates this fact.



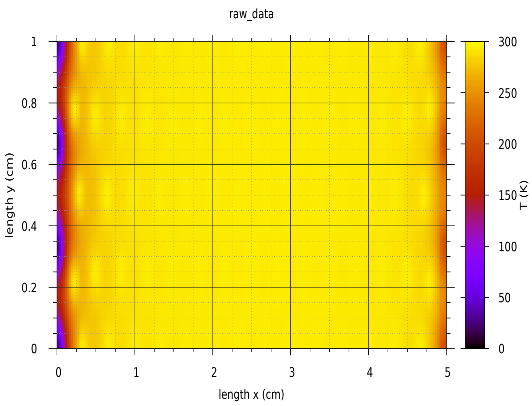
(a) Decay of the eigenvalues by a logarithmic scale on the y-axis. Each next mode is approximately one order of magnitude smaller.



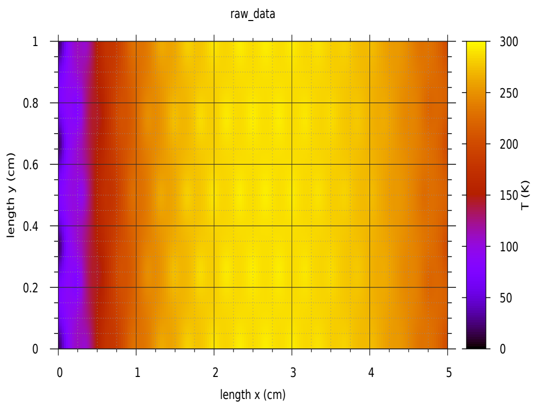
(b) The first 8 modes are plotted against the accummulated partial energy. The more modes are taken into account the more they must add up to 1. Here, only 2 modes are sufficient to capture more than 99 % of the systen.

Fig. 1.2

With this reasonable data equipped one can make now the test if these modes do actually yield the input data.



(a) Temperature distribution after 1 time step.

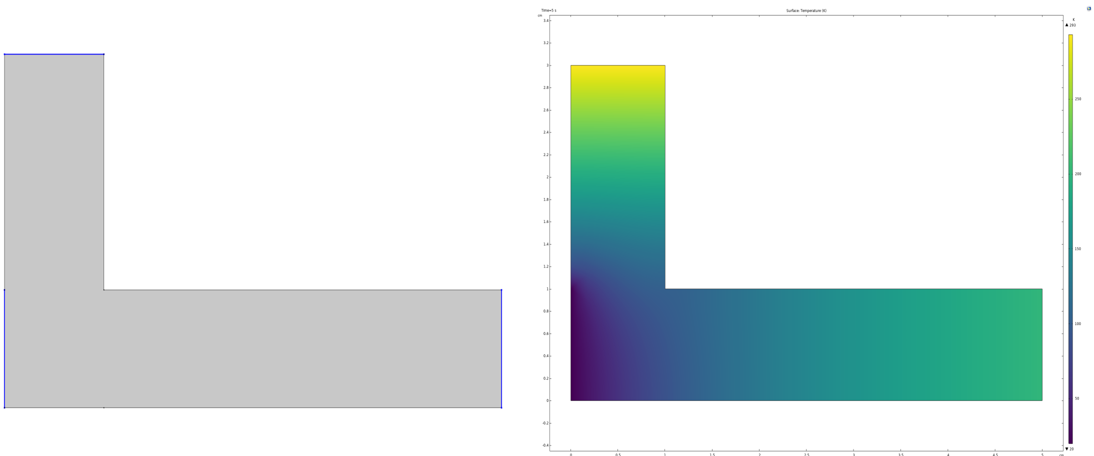


(b) Temperature distribution after 5 time steps.

Fig. 1.3: The temperature distribution is gradually smoothing over the geometry.

1.2 L-shape

To apply the code on a simple more complex geometry a L was chosen with three different borders held constant. To generate a more diverse temperature distribution the fixed temperature values are $T_0 = 20\,\mathrm{K}$, $T_1 = 123\,\mathrm{K}$ and $T_2 = 200\,\mathrm{K}$. As in the previous case, ambient temperature was set to $T_{\mathrm{amb}} = 293.15\,\mathrm{K}$.

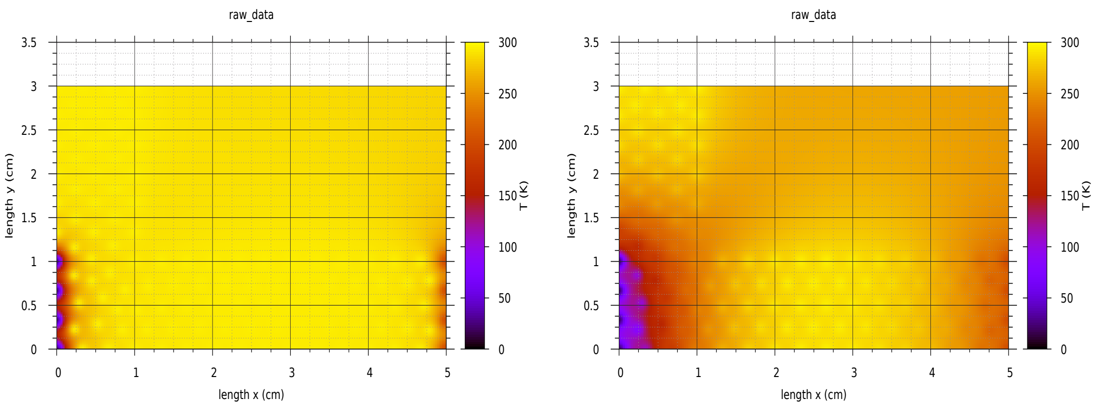


(a) Geometry setup with fixed temperature on the left T_0, on the right T_1 and on the top T_1.

(b) Temperature distribution after 5 s.

Fig. 1.4

As in the case with the I-shape the decay was of similar nature, i. e. only few eigenvalues are needed to take the total energy stored in the system. The code didn't need any modification at all, only different snapshot file from COMSOL was loaded. As in the previous case



(a) Temperature distribution after 1 time step.

(b) Temperature distribution after 5 time steps.

Fig. 1.5: The temperature distribution is slowly spreading through the geometry but it cannot be correct. As visible in the geometry the point $P(4,3)$ is obviously not part of the geometry but there is associated temperature to this point. That means that there is an error in postprocessing the data and remapping of the data values to the geometry must be done with a different approach. Mesh artefacts are tolerated at this time.

Next approach is not only export the data (nodes of mesh+values) but also the geometry from COMSOL. The nodes are not sufficient as artifacts will be created (here interpolation between nodes are performed and there is no naive way to “force“ interpolation only on provided nodes). The interpolator has no way to understand if a connection line of nodes lies inside or outside the geometry and is therefore doomed to fail. So another approach has to be taken. My next trials is to outsource the geometry and the mesh-generation (probably with Gmsh, generate snapshot matrix (probably with FreeFEM), generate POD marix (already done) and then to the interpolation of this matrix on the original geometry+mesh (not sure how to do). First trials were successful, i. e. geometry could be meshed and stationary Poisson-PDE could be solved (switching to time-dependent situation is next step but no fundamental problem) with FreeFEM. However, snapshot matrix needs to be extracted but should be possible.