# Simulation of heat conduction within ceramic blocks using finite elements and multigrid method

Bastian Pentenrieder

JASS, Saint Petersburg, Russia – April 7th, 2005

**Abstract**

Ceramic blocks are bricks of complicated geometry: They consist of several air cavities and the brick material itself. In order to investigate their behavior with respect to heat conduction, Fourier's PDE is considered for the steady-state case: $\operatorname{div}(\lambda \nabla T) = 0$, where $T$ denotes the temperature. The thermal conductivity $\lambda$ is a piecewise-constant function that assumes only two different values according to air and stone. Micro-convection and cavity radiation are both neglected within the model.

As a finite element (FE) discretization, we take a hierarchical system of squares with bilinear ansatz functions on the individual elements. Nodes that belong not only to the finest grid level exhibit more than one degree of freedom, i.e. for the representation of the solution we use a generating system instead of a basis. The linear equations resulting from the FE approach are solved by a special kind of multigrid algorithm that performs relaxed Jacobi iterations on all levels of the grid simultaneously. Due to the jumps in the coefficient $\lambda$, the crucial point is to determine the correct diagonal elements of the *intermediate* stiffness matrices for the coarser levels. They are essential for good convergence properties.

Obtaining the temperature distribution from the solution of the PDE, we compute the so-called effective thermal conductivity of the whole ceramic block and visualize the internal heat flux in some figures.

It is important to note that all the computations presented here can be fed with data through an interface, that was developed particularly for that purpose. This dialog allows the user to choose between different geometries, set the thermal conductivities and boundary conditions, adjust parameters for the algorithm and finally to start the calculations. The visualization afterwards is controlled through another interface window. In that way, even users without a strong background in the underlying mathematics can perform their own numerical experiments.

The algorithmic core of the program is based on *peano3d* – a software code which was written within the frame of a PhD thesis at the chair of Prof. Zenger from Technische Universität München. For the computation of the ceramic blocks, it had to be adjusted and extended mainly because of the non-constant coefficient $\lambda$.

In principle, all the ideas introduced here and especially the program itself work in three space dimensions. However, most of the illustrations, computations and results in this paper are restricted to the two-dimensional case.

# 1 Modelling

In scientific computing, the examination of a certain problem is always a three-stage process. Foremost, we need to think about how to model the task in the language of mathematics. Then, the next step is to discretize the obtained (continuous) equations and solve them by an appropriate numerical method. Finally, we have to present the results in an adequate way. This may be done by giving some key values or by applying visualization techniques to the computed dataset. According to this scheme, we start with introducing two fundamental laws of heat conduction that will help us to translate the ceramic block problem into mathematical equations:

By definition, heat is the energy that flows from the higher level of temperature to the lower (without any work being performed), whenever there exists a temperature gradient inside a body. If we know the internal temperature distribution for a given moment in time, we can calculate the *heat flux* $\dot{q}$ in every single point $\mathbf{x}$ by

$$\textit{Fourier's law:} \qquad \dot{q}(\mathbf{x}) = -\lambda \nabla T(\mathbf{x}) , \qquad [\lambda] = \frac{W}{mK} . \tag{1}$$

Note that Fourier's law gives us information about both direction and magnitude of the heat flux. The constant of proportionality $\lambda$ is called *thermal conductivity* and strongly depends on the material. In the absence of heat sources and sinks respectively, the diffusive heat entry into a volume increment $dV$,

$$\dot{Q}_{net} = \operatorname{div}(\lambda \nabla T)\, dV,$$

must be equal to the storage rate of thermal energy

$$\dot{H}_{st} = c_p\, dm\, \frac{\partial T}{\partial t} = \rho\, c_p\, \frac{\partial T}{\partial t}\, dV,$$

where

- $c_p$ is the *specific heat capacity* ($[c_p] = \frac{kJ}{kg\,K}$),
- $dm$ the mass of the volume $dV$,
- $\rho$ the density of the material and
- $\frac{\partial T}{\partial t}$ the partial derivative of the temperature with respect to time.

This simple balance of $\dot{Q}_{net}$ and $\dot{H}_{st}$ gives us

$$\textit{Fourier's PDE:} \qquad \rho\, c_p\, \frac{\partial T}{\partial t} = \operatorname{div}(\lambda \nabla T)\,, \tag{2}$$

which must be satisfied in every point $\mathbf{x}$ for every moment $t$. Since we only treat the steady-state case ($\frac{\partial T}{\partial t} = 0$) within the scope of this paper, the derived PDE of Fourier (2) reduces to

$$\operatorname{div}(\lambda(\mathbf{x}) \nabla T(\mathbf{x})) = 0 \qquad \forall \mathbf{x} \in \Omega\,. \tag{3}$$

The thermal conductivity is

$$\lambda(\mathbf{x}) = \begin{cases} \lambda_{brick} & \text{if } \mathbf{x} \in \Omega_{brick} \\ \lambda_{air} & \text{if } \mathbf{x} \in \Omega_{air} \end{cases} \qquad (\lambda_{brick} \gg \lambda_{air})\,. \tag{4}$$

Here, the set $\Omega$ denotes the whole ceramic block, while $\Omega_{air}$ and $\Omega_{brick}$ refer to the cavities and the brick material respectively (cf. figure 1). On the upper ($y = y_2$) and lower ($y = y_1$) part of the boundary, we prescribe Dirichlet conditions $T = T_i$ and $T = T_o$. They correspond to the temperature proportions

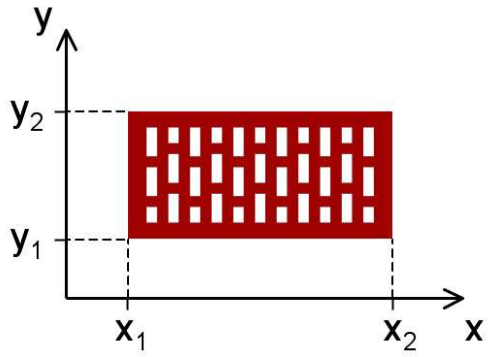


Figure 1: The ceramic block 1NF

inside and outside of an imaginary room. For reasons of symmetry, it makes sense to assume homogeneous Neumann boundary conditions

$$\frac{\partial T}{\partial n} = 0 \tag{5}$$

on the left ($x = x_1$) and right ($x = x_2$) hand side of the block. In expression (5), $n$ denotes the outward normal vector. The aim is to calculate the internal temperature distribution and hence

- the overall **heat transfer rate** through the ceramic block as well as
- the **effective thermal conductivity** which will be introduced later.

**Remark:** Using the model (3) and (4), we neglect all convection and radiation processes inside the ceramic block, whereas its thermal conductivity is represented by a piecewise-constant function, that exhibits jumps at the transitions $\partial\Omega_{brick} \cap \partial\Omega_{air}$ .

**Alternative models:** Assuming the air cavities to be a perfect heat insulation, a simpler approach would be

$$\mathrm{div}(\lambda_{brick} \nabla T(\mathbf{x})) = \lambda_{brick} \triangle T(\mathbf{x}) = 0 \qquad \forall \mathbf{x} \in \Omega_{brick}$$

with

$$\frac{\partial T}{\partial n}(\mathbf{x}) = 0 \qquad \forall \mathbf{x} \in \partial\Omega_{brick} \cap \partial\Omega_{air}$$

as boundary conditions at the (originally internal) transitions. Notice that in this case the domain on which the PDE shall be solved is just $\Omega_{brick}$ instead of $\Omega$.

# 2 Numerical computation

This section will explain how to transform problem (3) and (4) into the *weak formulation*, how we choose the finite element discretization for this one and finally how to solve the resulting system of linear equations using a special type of the multigrid method.

As an appropriate ansatz space we take

$$V := \{v \in H^1(\Omega) \text{ with } v = 0 \text{ on } \Gamma_D\},$$

where $H^1(\Omega)$ is the Sobolev space and $\Gamma_D$ denotes the Dirichlet part of the boundary. We multiply the original PDE (3) with an arbitrary element of V and integrate over the whole domain $\Omega$ :

$$\int_\Omega v \operatorname{div}(\lambda \nabla T)\, d\mathbf{x} = 0 \qquad \forall v \in V$$

*Green's formula* – which can be considered as a multidimensional integration by parts – helps us:

$$\begin{aligned} \int_\Omega v \operatorname{div}(\lambda \nabla T)\, d\mathbf{x} &= \int_{\partial\Omega} v\, \lambda \frac{\partial T}{\partial n}\, ds - \int_\Omega \lambda\, (\nabla T)^T (\nabla v)\, d\mathbf{x} \\ &= -\int_\Omega \lambda\, (\nabla T)^T (\nabla v)\, d\mathbf{x} \end{aligned}$$

The surface integral is equal to zero due to the homogeneous Neumann boundary conditions and the fact that we demand the test function $v$ to vanish on those part $\Gamma_D$ of the boundary where the Dirichlet boundary conditions have been declared.

Now we are able to rewrite the original problem in the following way: Find a function $T \in T_{Dir} + V$, so that

$$\int_\Omega \lambda \, (\nabla T)^T (\nabla v) \, d\mathbf{x} = 0 \qquad \forall v \in V \tag{6}$$

The shift of the ansatz space $V$ by $T_{Dir}$ is necessary because of the inhomogeneity of the Dirichlet boundary condition. The function $T_{Dir}$ is of the form

$$T_{Dir}(x, y) = \alpha y + \beta.$$

Figure 2 shows how it looks like. We will use exactly this function as an initial guess in the iterative solver later.

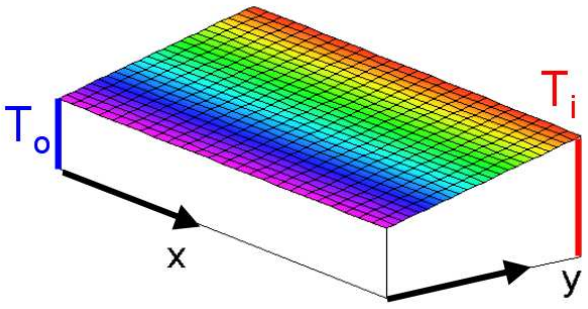


Figure 2: The function $T_{Dir}$

Since the ansatz space $V$ is an infinite-dimensional functional space, the next step is to replace it by a finite one. To this end, we lay a fine grid $\Omega^h$ of squares over the domain $\Omega$. Within each single element (square), bilinear ansatz functions are taken. The local node numbering and the graphs of the corresponding *shape functions* are depicted in figure 3. This particular FE approach results in the following element stiffness matrix:

$$A_{ij}^{(e)} = \int_{(e)} \lambda^{(e)} \, (\nabla \phi_j)^T (\nabla \phi_i) \, d\mathbf{x}$$

$$\Rightarrow \quad A^{(e)} = \frac{\lambda^{(e)}}{6} \begin{bmatrix} 4.0 & -1.0 & -1.0 & -2.0 \\ -1.0 & 4.0 & -2.0 & -1.0 \\ -1.0 & -2.0 & 4.0 & -1.0 \\ -2.0 & -1.0 & -1.0 & 4.0 \end{bmatrix}, \tag{7}$$

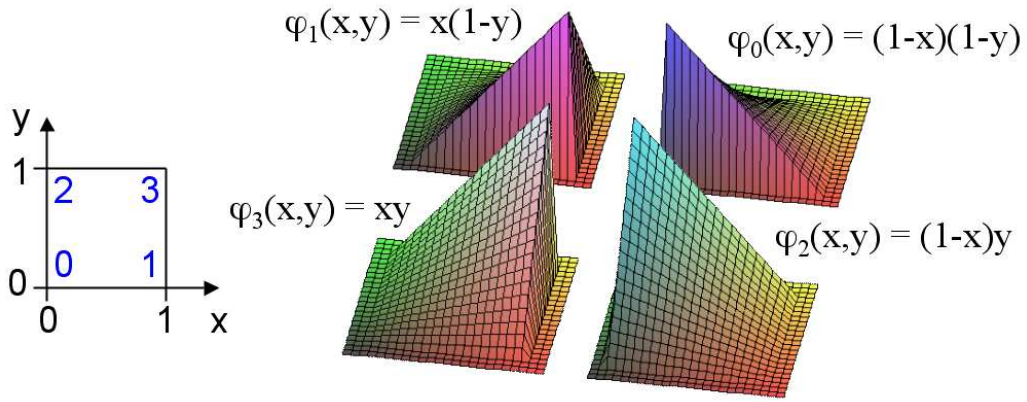


Figure 3: Bilinear ansatz functions on squares

where we have supposed that the mesh is sufficiently fine so that it resolves the material transitions accurately. The thermal conductivity inside a cell shall be constant: $\lambda = \lambda^{(e)}$.

**Remark:** If we write the temperatures at the corners of an element – in accordance with the introduced node numbering (see figure 3) – into a 4-vector $T^{(e)}$ and calculate the product $A^{(e)}\,T^{(e)}$, the individual components of the resulting 4-vector can be interpreted as heat flows from the respective nodes into the cell. On the other hand, the term $A^{(e)}\,T^{(e)}$ represents the contribution from this element to the residuals of the four degrees of freedom in its corners. The overall residual of a node represents the net heat flow from this node into its surroundings and is put together with the aid of the cell residuals from the four neighboring elements (cf. figure 4). Since we are looking for the steady-state temperature distribution, the iterative method should ideally converge towards a state where all the net heat flows (i.e. residuals) disappear.

**Sketch of the algorithm:** Besides the fine grid $\Omega^h$ mentioned above we introduce coarser grids $\Omega^{3h}, \Omega^{9h}, \ldots$ Each coarse grid cell consists of nine cells that belong to the next finer level (this ratio of partitioning is due to technical reasons). Notice that a temperature node of a coarse grid cell is a degree of freedom on all finer cells beneath. The algorithm works in the following way:

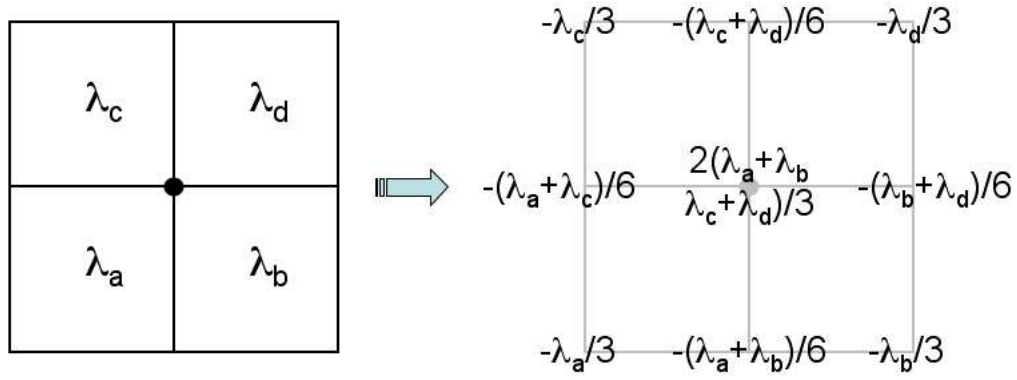


Figure 4: The overall residual in a node is assembled by the four corresponding cell residuals. Notice the weighting factors for the temperature nodes and their accordance with the entries in the element stiffness matrix.

1. On the coarsest grid level, we start the process of building up the current approximation to the solution using the temperature values from the initial guess $T_{Dir}$ or a previous iteration.

2. Then we recursively go to the next finer grid level by performing interpolation and adding so-called hierarchical surpluses, until we have reached the finest level $\Omega^h$. Figure 5 shows the principle in one space dimension. When we have descended to the lowest level, the built-up function equals the current iterate.

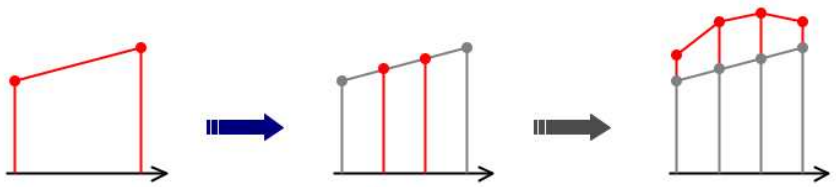

Figure 5: Interpolation (blue) and adding of hierarchical surpluses (grey)

3. On the mesh $\Omega^h$, the residual $\text{res}_h^{(k)}$ is computed and assembled ($k$ denotes the current iterate).

4. Restrict the residuals recursively to the coarser grids by using the weighting factors in figure 6.

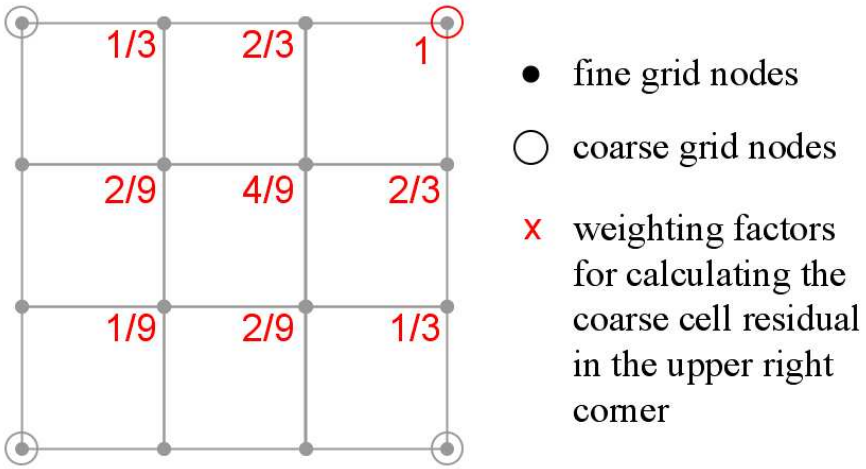


Figure 6: Weighting factors for restriction to the upper right coarse grid node

5. Perform an iteration step of the relaxed Jacobi method on all levels simultaneously: $T_{lh}^{(k+1)} = T_{lh}^{(k)} - D^{-1}\ \mathrm{res}_{lh}^{(k)}$ $(l = 1, 3, 9, 27, \ldots)$

The matrix $D^{-1}$ is the inverse diagonal of the system matrix. On the finest grid level, the individual components of $D$ are $2/3 \sum_{m=1}^{4} \lambda_m$, where $\lambda_1, \ldots, \lambda_4$ denote the thermal conductivities of the four cells surrounding the corresponding node. On coarser grid levels, it is not obvious how to obtain $D$, since the elements there in general consist of fine grid cells with different thermal conductivities. For this reason, we shall now consider the computation of the diagonal elements for the coarser grid levels in some detail. Since we are interested in the principle only, figure 7 shows a coarse grid cell that is partitioned into four – and not nine – fine grid cells. The respective thermal conductivities are denoted $a, b, c$ and $d$. Substituting these quantities for $\lambda^{(e)}$ into (7), we obtain the single element stiffness matrices. An *intermediate* stiffness matrix belonging to the coarse level is now computed by compiling the four element stiffness matrices into a 9-by-9 matrix $B$ (according to the fine grid node numbering) and multiplying this one with the restriction and interpolation operator respectively:

$$A^{(coarse)} = RBI, \tag{8}$$

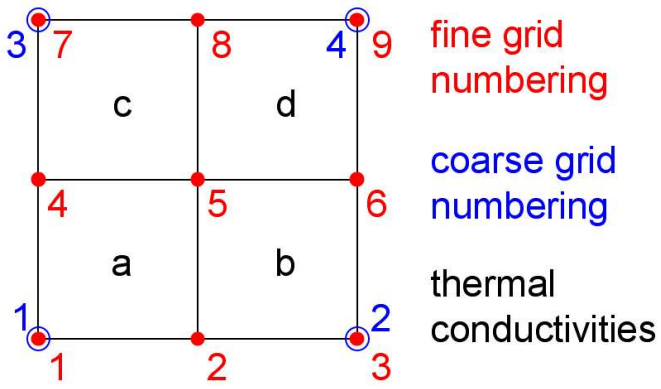


Figure 7: Fine and coarse grid node numbering

where

$$R = \begin{bmatrix} 1.0 & 0.5 & 0.0 & 0.5 & 0.25 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.5 & 1.0 & 0.0 & 0.25 & 0.5 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.5 & 0.25 & 0.0 & 1.0 & 0.5 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.25 & 0.5 & 0.0 & 0.5 & 1.0 \end{bmatrix}$$

and $I = R^T$. The motivation for (8) is provided by multigrid theory (cf. the excellent introductory textbook [Briggs 87]). Now that we have obtained the 4-by-4 matrix $A^{(coarse)}$, we are able to proceed to the next coarser grid level, this time taking $A^{(coarse)}$ instead of (7) as a starting-point. This procedure is repeated until we have reached the coarsest grid level. Afterwards, all the diagonal elements required in the correction steps are known. Note that it is sufficient to determine them once and for all during an initial run.

# 3 Results

Computations were carried out for three different geometries. They are depicted in figure 8. As thermal conductivities we took $\lambda_{air} = 0.025 \frac{W}{mK}$ and $\lambda_{brick} = 0.61/0.43 \frac{W}{mK}$. The room temperature was chosen to be constant at 20 degrees centigrade for all calculations, whereas the outside temperature altered in steps of 5 from +5 to +15.
We are interested in the overall heat transfer rate through the blocks:

$$\dot{Q} = -\int_{x_1}^{x_2} \lambda \frac{\partial T}{\partial y}(\xi, y_1)\, d\xi \tag{9}$$

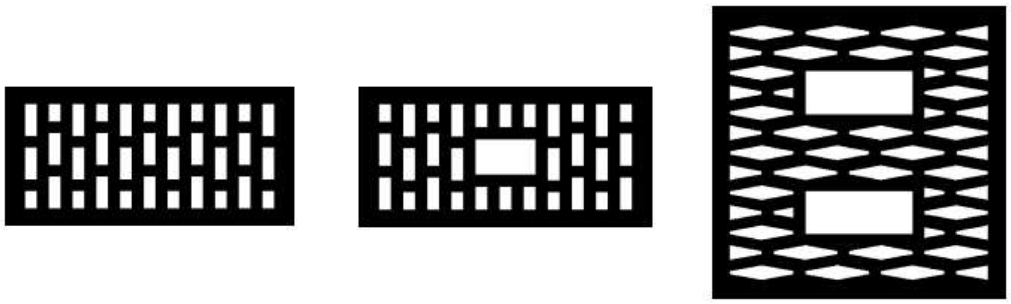

Figure 8: The ceramic blocks 1NF, 2NF and 4NF (from left to right)

Because $T$ is a steady-state temperature distribution, it doesn't make any difference at which y-position we integrate, i.e. instead of $y_1$ we could also take $y_2$ for evaluation (cf. figure 1). The effective thermal conductivity of a block is computed by

$$\lambda_{eff} = \dot{Q}\,\frac{y_2 - y_1}{T_o - T_i}\,.$$

It turned out that the computations deliver the same values $\lambda_{eff}$ independent from the room temperature $T_i$. This is exactly what the theory of heat transfer predicts. The table below shows the results:

| $\lambda_{eff}$ | $\lambda_{brick} = 0.61$ | $\lambda_{brick} = 0.43$ |
|---|---|---|
| $1NF$ | 0.405 | 0.29 |
| $2NF$ | 0.37 | 0.27 |
| $4NF$ | 0.20 | 0.15 |

# Literatur


[Braess 03] Dietrich Braess, *Finite Elemente, Theorie, schnelle Löser und Anwendungen in der Elastizitätstheorie; 3., korrigierte und ergänzte Auflage*, Springer, Berlin, 2003

[Briggs 87] William L. Briggs, A Multigrid Tutorial, printed for the Society for Industrial and Applied Mathematics by Lancaster Press, Pennsylvania, 1987

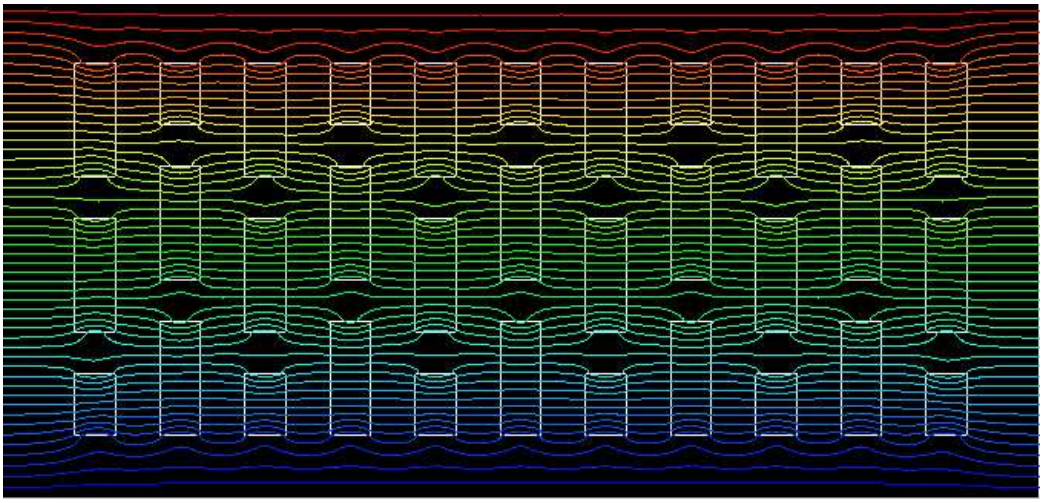

Figure 9: Isothermal lines colored according to their temperature level

[Incropera/DeWitt 02] Frank P. Incropera, David P. DeWitt, *Fundamentals of Heat and Mass Transfer, 5th edition*, John Wiley & Sons Ltd, Chichester, 2002

[Pögl 04] Markus Pögl, *Entwicklung eines cache-optimalen 3D Finite-Element-Verfahrens für große Probleme*, Dissertation, Institut für Informatik der TU München, 2004

[Slavyanov et al. 04] В. Грикуров, С. Славянов, Р. Трепков, Расчёт теплопроводящих характеристик керамических блоков с внутренними полостями, paper, Department of Mathematics and Computational Physics, Saint-Petersburg State University, 2004

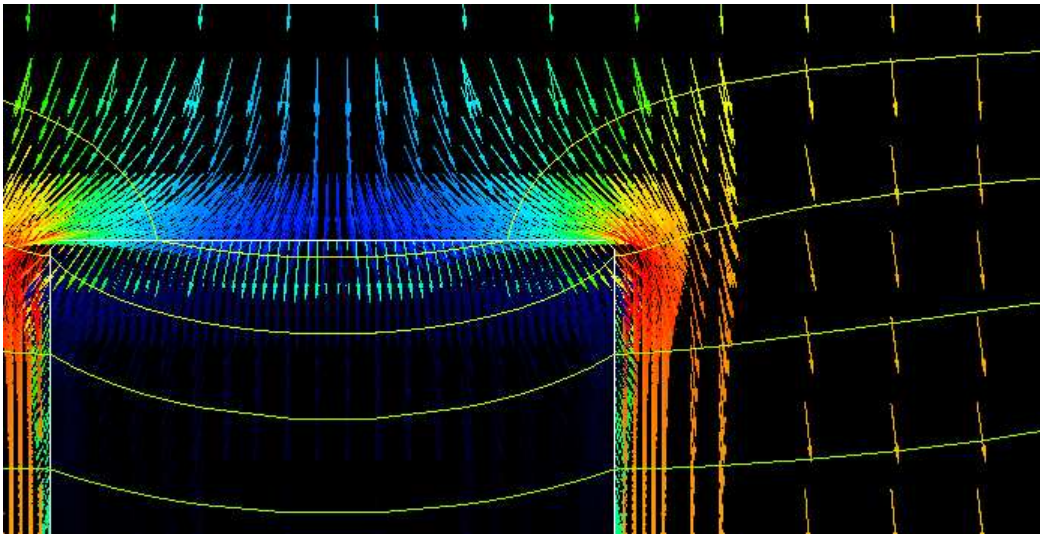

Figure 10: Detailed view of heat flux vectors. Their length has been standardized. Information about the magnitude is given by color. Notice that the arrows are perpendicular to the isothermal lines. The grid has been refined near the material transition.

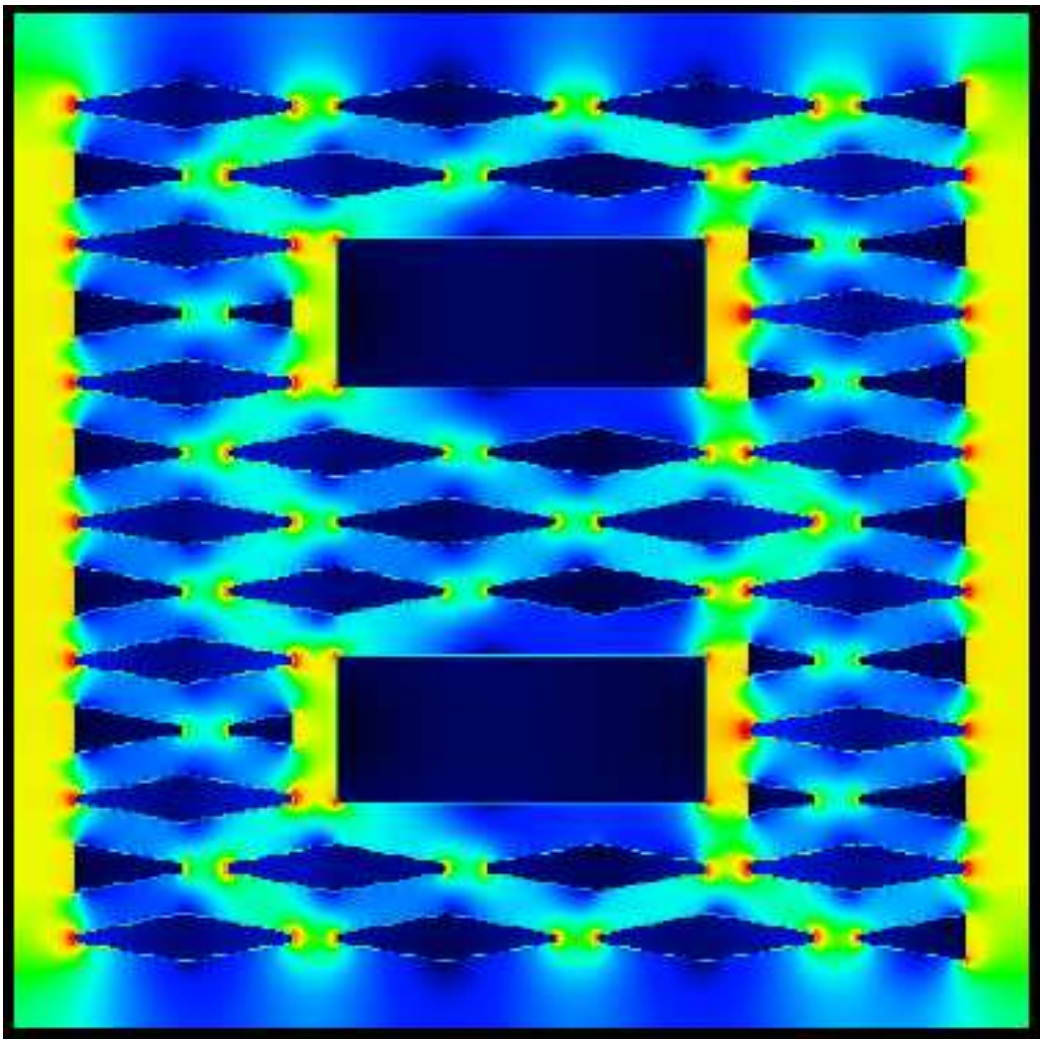

Figure 11: Heat flux within block 4NF as color map