

# An Object-Oriented Library for Heat Transfer Modelling and Simulation in Open Cell Foams<sup>★</sup>

Tobias M. Scheuermann<sup>\*</sup>, Paul Kotyczka<sup>\*</sup>,  
Christian Martens<sup>\*</sup>, Haithem Louati<sup>\*\*</sup>, Bernhard Maschke<sup>\*\*</sup>,  
Marie-Line Zanota<sup>\*\*\*</sup>, Isabelle Pitault<sup>\*\*</sup>

<sup>\*</sup> *Technical University of Munich, Department of Mechanical Engineering, Chair of Automatic Control, Boltzmannstraße 15, 85748 Garching, Germany.*

<sup>\*\*</sup> *Univ. Lyon, Université Claude Bernard Lyon 1, CNRS, LAGEPP UMR 5007, 43 boulevard du 11 novembre 1918, F-69100 Villeurbanne, France.*

<sup>\*\*\*</sup> *Univ. Lyon, CNRS, CPE Lyon, UCBL, LGPC UMR 5285, 43 boulevard du 11 novembre 1918, F-69100 Villeurbanne, France.*

---

## Abstract:

Metallic open cell foams have multiple applications in industry, e.g. as catalyst supports in chemical processes. Their regular or heterogeneous microscopic structure determines the macroscopic thermodynamic and chemical properties. We present an object-oriented python library that generates state space models for simulation and control from the microscopic foam data, which can be imported from the image processing tool iMorph. The foam topology and the 3D geometric data are the basis for discrete modeling of the balance laws using the cell method. While the material structure imposes a primal chain complex to define discrete thermodynamic driving forces, the internal energy balance is evaluated on a second chain complex, which is constructed by topological duality. The heat exchange between the solid and the fluid phase is described based on the available surface data. We illustrate in detail the construction of the dual chain complexes, and we show how the structured discrete model directly maps to the software objects of the python code. As a test case, we present simulation results for a foam with a Kelvin cell structure, and compare them to a surrogate finite element model with homogeneous parameters.

*Keywords:* Port-Hamiltonian systems, metallic foam, cell method, distributed parameter systems, discrete modeling, geometric discretization, process systems, simulation

---

## 1. INTRODUCTION

Metallic foams are a type of material that is used in multiple ways for industrial purposes. Two classes of metallic foams are distinguished: closed and open cell foams. In closed cell foams, the fluid phase is encapsulated in closed cavities inside the foam. Open cell foams have connected porous cells so that the fluid can flow through the material. In this paper, we will concentrate on the latter. Due to their high surface to volume ratio, open cell metallic foams are used in catalytic reactors, see e.g. Frey et al. (2016). In order to design and control the chemical processes in a reactor, numerical models of the thermodynamic behaviour are needed. Existing approaches use effective properties, e.g. from volume averaging over Cartesian unit cells (Quintard et al., 1997).

With the use of tomography, precise 3D voxel data of a given foam sample can be generated and topological

as well as geometric data can be extracted using image processing software like iMorph (Brun et al., 2008). We will show an approach to set up a numerical model for the heat transfer on open cell foams that is directly based on the possibly heterogeneous foam topology. Microscopic material parameters and the exact geometry complete the model in the discrete constitutive equations.

The separation of a (Dirac) interconnection structure to describe the structural exchange of power (or the time derivative of another appropriate potential) via pairs of conjugated port variables from material-dependent constitutive equations and energy storage, is at the heart of the Port-Hamiltonian (PH) framework, see e.g. Duindam et al. (2009) for an overview. In Seslija et al. (2014), and later for non-uniform boundary conditions in Kotyczka and Maschke (2017), the discrete modelling of conservation laws on dual chain complexes was presented. The preliminary work (Scheuermann et al., 2019) illustrates the discrete modelling of heat transfer and exchange on open cell foams.

<sup>★</sup> This work was supported by Deutsche Forschungsgemeinschaft (project number 317092854) and Agence Nationale de la Recherche (ID ANR-16-CE92-0028), project DFG-ANR INFIDHEM.

The main contributions of this paper are (a) necessary extensions for the classification of topological objects from the regular 2D case as presented in Kotyczka and Maschke (2017) to irregular 3D meshes, (b) the consideration of heat exchange between two phases and (c) the objected-oriented model generation in python.

In Section 2, we give some background on image processing of tomographic foam pictures. Section 3 shows the structured representation of the coupled heat equation on dual complexes, while Section 4 presents the immediate mapping of the model to python code. A numerical example is given in Section 5, and the paper closes with final remarks and an outlook in Section 6.

## 2. IMAGE PROCESSING

The input data for model generation and simulation is obtained from the image processing tool iMorph<sup>1</sup>. iMorph can extract the structure of the foam from 3D tomography pictures. A typical example is shown below. Fig. 1a shows the image of an open cell foam sample, while Fig. 1b displays the extracted solid graph.

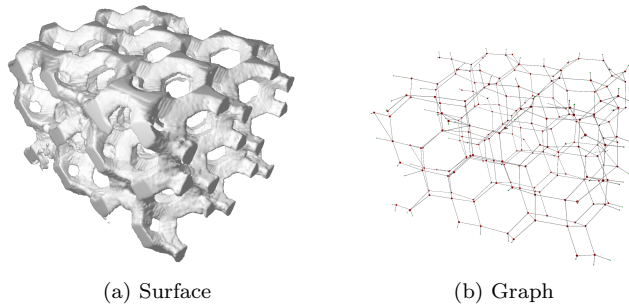


Fig. 1. Metallic open cell foam (Kelvin cells)

Besides the solid nodes (Fig. 2a) and struts (Fig. 2b), which are represented by the edges of the solid graph, iMorph identifies cells (Fig. 2c) in the fluid phase. These cells are connected by so-called “windows” (Fig. 2d).

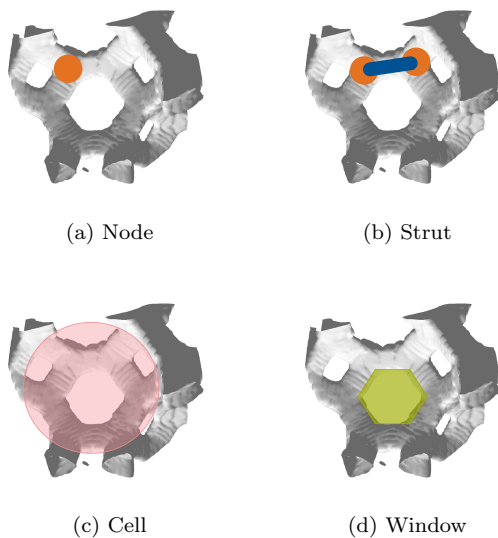


Fig. 2. Structures in open cell foams defined by iMorph

<sup>1</sup> <http://imorph.sourceforge.net/>

## 3. DISCRETE HEAT EQUATION ON DUAL COMPLEXES

The modelling is based on the cell method, see Alotto et al. (2013) for an introduction to this numerical scheme with references to the original works (Tonti, 2001) and applications. The PH framework explicitly considers open systems, i.e. systems with boundary energy flow, see Seslija et al. (2014) for the discrete modelling of conservation laws and van der Schaft and Maschke (2013) for PH systems on graphs. We follow the regular 2D approach described in Kotyczka and Maschke (2017). The heterogeneous 3D case considered here requires some adaptations and additions, which are illustrated below.

### 3.1 Cells, Chains and Chain Complex

The topology and geometry of the foam is described in a structured way using  $j$ -dimensional cells, or in short “ $j$ -cells”<sup>2</sup>, see Arnold (1989), Section 35.D or Flanders (1989), Section 5.5. A  $j$ -cell is a geometric object that consists of a convex polyhedron  $D \subset \mathbb{R}^j$ , a differentiable map  $f : D \rightarrow M$  on the  $n$ -dimensional manifold  $M$  and an orientation. A formal sum of  $j$ -cells is called  $j$ -chain.

The linear vector space of  $j$ -chains on a tessellation  $K$  is denoted  $C_j(K, \mathbb{R})$ . The boundary of each  $j$ -cell consists of a  $j - 1$ -chain and is found by applying the boundary operator  $\partial_j$ . Applying the boundary operator twice to a  $j$ -chain results in an empty set, which is the central property of a chain complex, see e.g. Jänich (2001), Section 7.6. The spaces of  $j$ -chains,  $j = n, \dots, 0$ , which, connected via the boundary operators, form an  $n$ -complex, can be represented in a sequence diagram:

$$C_n(K, \mathbb{R}) \xrightarrow{\partial_n} C_{n-1}(K, \mathbb{R}) \xrightarrow{\partial_{n-1}} \dots \xrightarrow{\partial_1} C_0(K, \mathbb{R}) \quad (1)$$

In the following, we call a  $n$ -chain with the collection of all  $j$ -cells,  $j = 0 \dots n$ , appearing in the sequence above, an  $n$ -complex. The symbol  $\partial$  will be used for both the boundary operator and its matrix representation, i.e. an incidence matrix. For our application, only the case with  $n = 3$  is relevant, so we will restrict ourselves to this case.

### 3.2 Definition of the Primal 3-Complex

The primal 3-complex is initially given by the structure of the solid phase. Since an  $n$ -complex can be seen as a generalized *directed* graph, orientations have to be assigned to all  $j$ -cells. The nodes (0-cells) and edges (1-cells) of the primal 3-complex can be taken directly from the graph generated with iMorph. Faces (2-cells) correspond to the iMorph windows. The windows that enclose a fluid cell define a volume (3-cells).

The following classification of inner and border  $j$ -cells is necessary for the direct imposition of boundary conditions in the numerical model. To realize Neumann boundary conditions (NBCs), i.e. heat flux boundary conditions on

<sup>2</sup> The term “cell” is used in two contexts, that should not be confused with each other: It is used in iMorph to describe a cavity in the foam or a  $j$ -dimensional geometric object. Therefore, the latter is always denoted as  $j$ -cell.

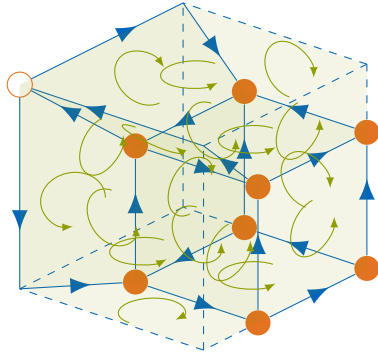


Fig. 3. Primal complex

the appropriate dual objects, see Subsection 3.3, additional border nodes must be defined, which lead to additional edges, faces and volumes. “Additional” in the sense, that their co-chains will not be part of the system dynamics, but they appear in the definition of the collocated output.

Fig. 3 shows a minimal example for a 3-complex with the orientation of the  $j$ -cells. The different categories of  $j$ -cells are described below.

*Inner nodes:* Solid nodes inside the domain are called inner nodes and are denoted by  $n_i \in \mathcal{N}_i$ .

*Border nodes:* Solid nodes on the boundary are called border nodes and are denoted by  $n_b \in \mathcal{N}_b$ . At these nodes, a Dirichlet boundary condition (DBC) is imposed.

*Additional border nodes:* These nodes, denoted by  $n_b \in \mathcal{N}_B$ , are not a representation of a solid node, but an intersection of a strut with the boundary (border edge, see below). Through the dual face to this edge, see next subsection, a NBC is imposed.

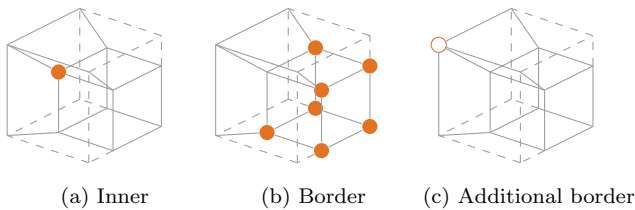


Fig. 4. Primal nodes

*Inner edges:* Inner edges  $e_i \in \mathcal{E}_i$  are connections of the inner nodes and border nodes. They represent struts that are entirely inside the domain or on its boundary ( $n_i$  with  $n_i$ ,  $n_b$  with  $n_b$  and  $n_i$  with  $n_b$ ).

*Border edges:* Border edges  $e_b \in \mathcal{E}_b$  connect inner nodes to additional border nodes and represent struts that cross the system boundary.

*Additional border edges:* These edges  $e_B \in \mathcal{E}_B$  have no representation in the solid graph, but they are necessary to fill the entire domain with volumes.

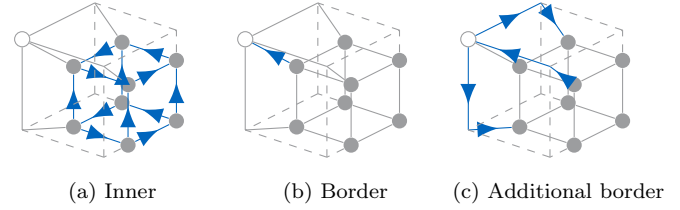


Fig. 5. Primal edges

*Inner faces:* Faces belonging to windows that are entirely inside the domain, are called inner faces  $f_i \in \mathcal{F}_i$ .

*Border faces:* Border faces  $f_b \in \mathcal{F}_b$  belong to windows that are not completely inside the domain.

*Additional border faces:* These faces  $f_B \in \mathcal{F}_B$  do not belong to a window, but they are necessary to fill the entire domain with volumes.

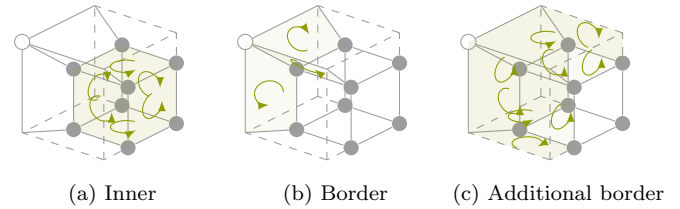


Fig. 6. Primal faces

*Inner volumes:* Volumes that lie inside the domain or on the boundary with a DBC are inner volumes  $v_i \in \mathcal{V}_i$ .

*Border volumes:* Only volumes at the boundary with a NBC are border volumes  $v_b \in \mathcal{V}_b$ .

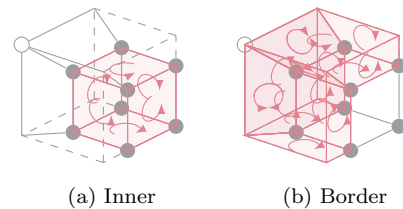


Fig. 7. Primal volumes

*Remark 1.* On first sight it may seem, that some nodes in Fig. 3, especially at the corners, are missing. However, they were left out intentionally. Similar to the additional boundary edges in the 2D case in (Kotyczka and Maschke, 2017, Fig. 5), that have no nodes at the corner of the face, the volumes in the 3D case can also have corners without nodes. In 3D, there can even be kinks in the faces without having a “real” edge at that position. These kinks are drawn with dotted lines and lie *inside* a face and have therefore no effect on the result of the *boundary* operator applied to the face.

*Remark 2.* The categorization differs from Kotyczka and Maschke (2017), because we choose to evaluate the energy balance on the dual instead of the primal  $n$ -cells, which makes it possible to locate the discrete temperatures at the solid nodes. Subsequently, the driving force is evaluated on the primal, i. e. the material edges.

### 3.3 Construction of the Dual 3-Complex

The dual 3-complex is defined by construction. For better visibility, only one dual  $j$ -cell is drawn in Fig. 8. The same procedure is repeated for all other primal  $j$ -cells.

A barycentric dual is used, as in Alotto et al. (2013). This means, that the dual node is located at the barycentre of the primal volume (Fig. 8a). Accordingly, a dual edge intersects with its primal face at the barycentre of the face (Fig. 8b) and the dual face intersects with the primal edge also at the barycentre of the edge (Fig. 8c). The dual complex is completed with the dual volumes around the primal nodes (Fig. 8d).

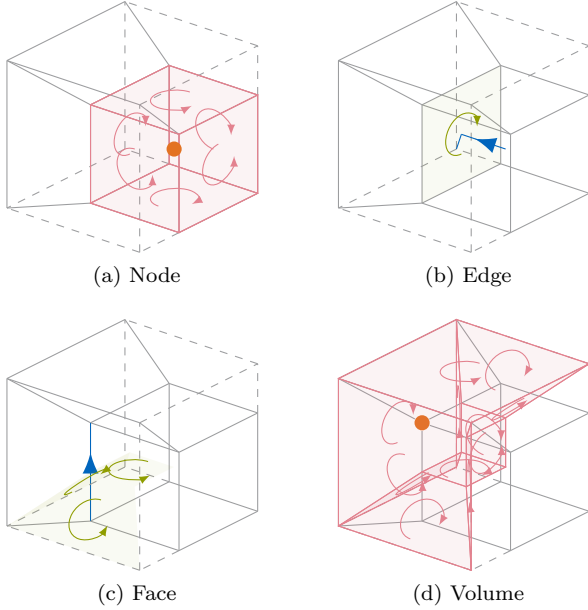


Fig. 8. Primal and associated dual cells

### 3.4 Discrete PH Representation

For a structured discrete model of the heat transfer on the foam, we start with the well-known heat equation with distributed parameters on a single phase,  $\mathbf{x} \in \Omega \subset \mathbb{R}^3$ ,  $t \in \mathbb{R}_0^+$ ,

$$c\dot{T}(\mathbf{x}, t) = \lambda \Delta T(\mathbf{x}, t). \quad (2)$$

$T(\mathbf{x}, t)$  denotes the temperature, the heat capacity  $c$  and the thermal conductivity  $\lambda$  are assumed to be constant. We rewrite (2) in port-Hamiltonian form (neglecting for the moment the boundary conditions) using the inner energy density  $u(\mathbf{x}, t)$  as state and  $T(\mathbf{x}, t)$  as co-state/effort<sup>3</sup>,

$$\begin{bmatrix} \dot{u} \\ \mathbf{f} \end{bmatrix} = \begin{bmatrix} 0 & -\text{div} \\ -\text{grad} & \mathbf{0} \end{bmatrix} \begin{bmatrix} T \\ \phi \end{bmatrix}. \quad (3)$$

$\phi(\mathbf{x}, t)$  and  $\mathbf{f}(\mathbf{x}, t)$  denote the vectors of heat flux and the temperature gradient as the thermodynamic driving force. The model is completed with the constitutive laws

$$\phi = \lambda \mathbf{f} \quad u = cT. \quad (4)$$

The discrete model is found by integrating the equations over the appropriate  $j$ -chains of the dual and the primal complex, respectively, as indicated in Table 3 with superscript s or f referring to the solid or the fluid phase.

<sup>3</sup> Which is the conjugate quantity w. r. t. the artificial potential  $\int_{\Omega} \frac{1}{c} u^2(\mathbf{x}, t) d\mathbf{x}$ .

Table 1.  $j$ -chains and associated quantities

$j$ -chain	(Integral) physical quantity
Primal node $n_k$	Temperature $T_k^{s/f}$
Primal edge $e_k$	Driving force (temperature difference) $F_k^{s/f}$
Dual face $\hat{f}_k$	Heat flow rate $\hat{\Phi}_k^{s/f}$
Dual volume $\hat{v}_k$	Energy $\hat{U}_k^{s/f}$

If  $k$  is the index for a dual control volume, and the set  $\mathcal{I}(k)$  contains the indices of the boundary faces, the discrete energy balance on such a control volume can be written for both the solid and the fluid phase as

$$\frac{\partial}{\partial t} \hat{U}_k^s = - \sum_{l \in \mathcal{I}(k)} \hat{\Phi}_{k,l}^s - \hat{\Phi}_k^{sf} \quad (5a)$$

$$\frac{\partial}{\partial t} \hat{U}_k^f = - \sum_{l \in \mathcal{I}(k)} \hat{\Phi}_{k,l}^f + \hat{\Phi}_k^{sf}. \quad (5b)$$

The heat flow  $\hat{\Phi}_k^{sf}$  represents the heat transfer between both phases. The temperature differences along a strut (index  $k$ , 1 and 2 refer to the start and end node) for both phase, as well as between both phases are

$$F_k^s = -(T_{k,2}^s - T_{k,1}^s), \quad F_k^f = -(T_{k,2}^f - T_{k,1}^f) \quad (6a)$$

$$F_k^{sf} = T_k^s - T_k^f. \quad (6b)$$

Finally, the discrete approximations of the constitutive equations (4) for both phases, together with the heat transfer model between both phases are

$$\hat{\Phi}_k^s = \frac{\lambda A_k^s F_k^s}{|\mathbf{r}_{k,2} - \mathbf{r}_{k,1}|}, \quad \hat{\Phi}_k^f = \frac{\lambda A_k^f F_k^f}{|\mathbf{r}_{k,2} - \mathbf{r}_{k,1}|} \quad (7a)$$

$$\hat{\Phi}_k^{sf} = \alpha A_k^{sf} F_k^{sf} \quad (7b)$$

$$\hat{U}_k^s = V_k^s c^s T_k^s, \quad \hat{U}_k^f = V_k^f c^f T_k^f. \quad (7c)$$

The discrete geometry parameters (note that (5) and (6) contain only topological information) are given in Table 2.

Table 2. Geometry parameters

Parameter	Definition
$\mathbf{r}_k$	Position vector of node $n_k$
$A_k^{s/f}$	Solid / fluid part of the area of $f_k$
$A_k^{sf}$	Contact area of the phases in $v_k$
$V_k^{s/f}$	Solid / fluid part of the volume of $v_k$

To obtain a numerical model of the heat transfer in the complete foam, we collect the whole set of variables  $\hat{U}_k$ ,  $F_k$ ,  $T_k$  and  $\hat{\Phi}_k$  in the vectors  $\hat{\mathbf{U}}_{i/b}$ ,  $\mathbf{F}_{i/b}$ ,  $\mathbf{T}_{i/b}$  and  $\hat{\Phi}_{i/b}$ , which represent inner / border co-chains as algebraically dual objects to the  $j$ -chains of the primal and the dual complex<sup>4</sup>.

The result is the following system of equations, where  $\hat{\mathbf{d}}_{ii/bi}^3 = -(\mathbf{d}_{ii/bi}^1)^T$  and  $\mathbf{d}_{ii/bi}^1$  denote the co-incidence matrices (i. e. the transposed boundary matrices) between

<sup>4</sup> For a given  $j-1$ -co-chain  $c^{j-1}$ , which contains the integral values of a quantity over  $j-1$ -chains, the duality pairing, see Seslija et al. (2014),

$$\langle c^{j-1}, \partial_j c_j \rangle = \langle d^j c^{j-1}, c_j \rangle \quad (8)$$

defines the co-boundary operator  $d^j$ . The sequence of spaces of co-chains and co-boundary operators defines a co-chain complex

$$C^0(K, \mathbb{R}) \xrightarrow{d^1} C^1(K, \mathbb{R}) \xrightarrow{d^2} \dots \xrightarrow{d^n} C^n(K, \mathbb{R}). \quad (9)$$

faces and volumes on the dual complex and nodes and edges on the primal complex, respectively<sup>5</sup>.

$$\begin{bmatrix} \hat{\mathbf{U}}_{if}^s \\ \hat{\mathbf{U}}_{if}^{sf} \\ \mathbf{F}_{if}^s \\ \mathbf{F}_{if}^{sf} \\ \mathbf{F}_i^s \\ \mathbf{F}_i^{sf} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & (-\mathbf{d}_{ii}^1)^T & \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & (-\mathbf{d}_{ii}^1)^T & -\mathbf{I} \\ \mathbf{d}_{ii}^1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{d}_{ii}^1 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -\mathbf{I} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{T}_{if}^s \\ \mathbf{T}_{if}^{sf} \\ \hat{\Phi}_{if}^s \\ \hat{\Phi}_{if}^{sf} \\ \hat{\Phi}_i^s \\ \hat{\Phi}_i^{sf} \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{0} & (-\mathbf{d}_{ib}^1)^T & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & (-\mathbf{d}_{ib}^1)^T \\ \mathbf{d}_{ib}^1 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{d}_{ib}^1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{T}_{ib}^s \\ \mathbf{T}_{ib}^{sf} \\ \hat{\Phi}_{ib}^s \\ \hat{\Phi}_{ib}^{sf} \end{bmatrix} \quad (10)$$

The subscripts  $i$  and  $b$  denote the locations (in the interior or at the boundary) of the  $j$ -chains, on which the discrete quantities are defined as presented in the previous subsections. Note that the skew-symmetry of the first matrix mimics the formal skew-adjointness of the matrix operator in (3).

The model is again completed by the constitutive laws

$$\hat{\mathbf{U}}^{s/f} = \mathbf{C}^{s/f} \mathbf{T}^{s/f} \quad (11a)$$

$$\hat{\Phi}^{s/f} = \mathbf{\Lambda}^{s/f} \mathbf{F}^{s/f} \quad (11b)$$

with the diagonal matrices

$$\mathbf{C}^{s/f} = \text{diag}(V_k^{s/f} c^{s/f}) \quad (12a)$$

$$\mathbf{\Lambda}^{s/f} = \text{diag} \left( \frac{\lambda A_k^{s/f}}{|\mathbf{r}_{k,2} - \mathbf{r}_{k,1}|} \right) \quad (12b)$$

#### 4. IMPLEMENTATION

For the implementation of the 3-complexes and their  $j$ -cells, we chose an objected oriented approach using the programming language `Python`. The goal of this implementation is to represent the relations between  $j$ -cells in the code. The general structure of the core classes is shown as a UML diagram in Fig. 9. For better treatment of  $j$ -cells with reverse orientation, the implementation includes some more classes than shown, but they follow the same architecture.

To avoid redundancy, the classes of all  $j$ -cells inherit from a `Cell` class where common properties like numbering or labeling are implemented. `Node`, `Edge`, `Face` and `Volumes` classes must be instantiated from top to bottom, since every class needs an aggregation of its predecessor. This approach relates to the application of the co-boundary operator as in (9).

All objects of  $j$ -cells are collected in an instance of the `PrimalComplex` class, where the classification is implemented and the incidence matrices are calculated. The `DualComplex` automatically generates all dual  $j$ -cells.

#### 5. NUMERICAL EXAMPLE

The presented approach is applied to a grid based on Kelvin cells as shown in Fig. 10, which has 848 degrees

<sup>5</sup> For the relations of co-incidence matrices between the dual complexes, see Kotyczka and Maschke (2017) or Seslija et al. (2014).

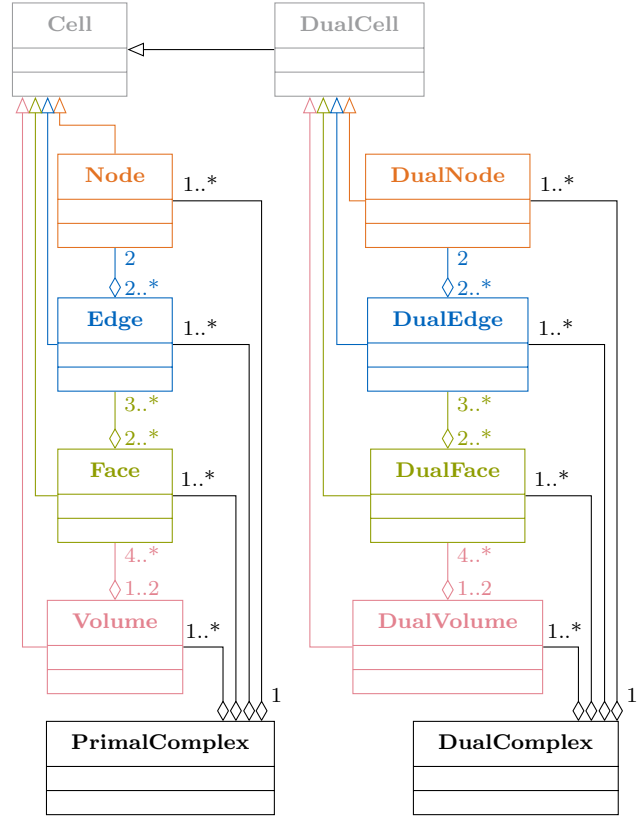


Fig. 9. Simplified UML diagram

of freedom (DOFs). For better replicability, the grid is constructed, so that we can test the numerical method without depending on user settings in `iMorph` or the need to compensate possibly occurring defaults in the `iMorph` result. On the top and bottom boundary, a DBC is applied (Fig. 11a). The other boundaries have a NBC (Fig. 11b), in our case the heat transfer is set to 0, meaning it is perfectly isolated at these boundaries.

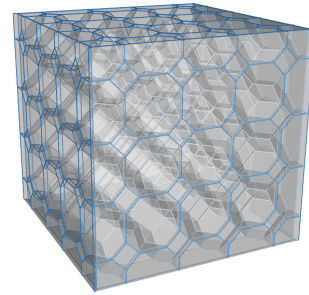
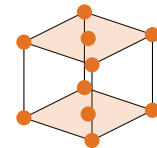
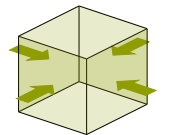


Fig. 10. Geometry of the example foam



(a) Dirichlet boundary condition (DBC)



(b) Neumann boundary condition (NBC)

Fig. 11. Primal volumes

The material parameters used in the simulation are given in Table 3.

Table 3. Material parameters

Dimensions	$l \times w \times h$	$40 \times 40 \times 40$	mm
Mass	$m$	16.463	g
Density of aluminium	$\rho^s$	$2.7 \times 10^{-3}$	$\frac{\text{g}}{\text{mm}^3}$
Density of air	$\rho^f$	$1.204 \times 10^{-6}$	$\frac{\text{g}}{\text{mm}^3}$
Heat capacity of Al	$c^s$	0.897	$\frac{\text{J}}{\text{g K}}$
Heat capacity of air	$c^f$	1.005	$\frac{\text{J}}{\text{g K}}$
Thermal conductivity of Al	$\lambda^s$	0.2	$\frac{\text{W}}{\text{mm K}}$
Thermal conductivity of air	$\lambda^f$	$2.6 \times 10^{-5}$	$\frac{\text{W}}{\text{mm K}}$
Heat transfer coefficient	$\alpha$	$1.0 \times 10^{-4}$	$\frac{\text{W}}{\text{mm}^2 \text{K}}$
Surrogate thermal diffusivity	$a_{\text{eff}}$	1.85	$\frac{\text{mm}^2}{\text{s}}$

Fig. 12 shows the transient behaviour of the temperature on 4 selected nodes.  $T_0$  is the constant temperature at the bottom boundary, while  $T_3$  is increased at the top.  $T_1$  and  $T_2$  are the temperatures of two nodes at different heights close to the front boundary.

For comparison, a Finite Element (FE) simulation with 18081 DOFs was performed with a surrogate parameter for the diffusivity  $a_{\text{eff}} = \frac{\lambda_{\text{eff}}}{\rho_{\text{eff}} c_{\text{eff}}}$  using FEniCS (Alnæs et al., 2015). The results are shown with markers and the superscript  $c$ .

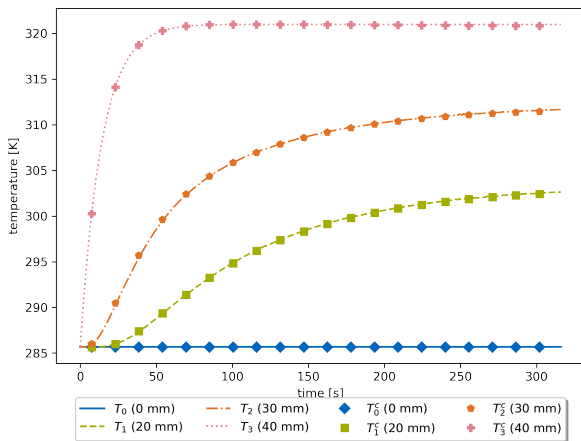


Fig. 12. Transient behaviour of the foam

The perfect matching of both our simulation based on the structured model with the surrogate FE simulation is due to two facts: (a) the surrogate diffusivity has been determined by curve fitting and (b) we considered the case with a quite homogeneous and regular distribution without material accumulation.

## 6. CONCLUSION AND OUTLOOK

We showed a structured approach to obtain a numerical model of heat transfer through metallic open cell foams, in which the separation of topology (expressed in terms of co-incidence matrices) on the one side and geometry and material parameters (constitutive equations) on the other side mimics the PH structure of the local PDE model. The model allows to identify macroscopic foam parameters,

and can be used for design optimization and (after possible model reduction) for control.

The model structure directly maps to the objects and dependencies of the object oriented python library, which can read topology and geometry data over an interface to the iMorph image processing software. We presented the simulation of a realistic foam model and its comparison to a FE simulation with surrogate effective parameters.

At the moment, we work in several directions: (a) the simulation of real foam data and comparison with the experimental data obtained at LGPC Lyon, (b) the integration of convection in the model and (c) improving robustness of our model generation with respect to artefacts like not fully connected graphs from image processing.

## ACKNOWLEDGEMENTS

The authors cordially thank Jérôme Vicente from University Aix-Marseille for the help with iMorph and in particular for implementing modifications in the new iMorph version that allow us to directly access all necessary objects and parameters.

## REFERENCES

- Alnæs, M.S., Blechta, J., Hake, J., Johansson, A., Kehlet, B., Logg, A., Richardson, C., Ring, J., Rognes, M.E., and Wells, G.N. (2015). The FEniCS project version 1.5. *Archive of Numerical Software*, 3(100). doi:10.11588/ans.2015.100.20553.
- Alotto, P., Freschi, F., Repetto, M., and Rosso, C. (2013). *The Cell Method for Electrical Engineering and Multiphysics Problems*. Springer Berlin Heidelberg. doi:10.1007/978-3-642-36101-2.
- Arnold, V.I. (1989). *Mathematical Methods of Classical Mechanics*. Springer New York. doi:10.1007/978-1-4757-2063-1.
- Brun, E., Vicente, J., Topin, F., and Occelli, R. (2008). Imorph: A 3d morphological tool to fully analyse all kind of cellular materials. *Cellular Metals for Structural and Functional Applications*.
- Duindam, V., Macchelli, A., Stramigioli, S., and Bruyninckx, H. (2009). *Modeling and Control of Complex Physical Systems*. Springer-Verlag GmbH.
- Flanders, H. (1989). *Differential Forms with Applications to the Physical Sciences*. Dover Publications Inc.
- Frey, M., Romero, T., Roger, A.C., and Edouard, D. (2016). Open cell foam catalysts for CO2 methanation: Presentation of coating procedures and in situ exothermicity reaction study by infrared thermography. *Catalysis Today*, 273. doi: 10.1016/j.cattod.2016.03.016.
- Jänich, K. (2001). *Vector Analysis*. Springer.
- Kotyczka, P. and Maschke, B. (2017). Discrete port-Hamiltonian formulation and numerical approximation for systems of two conservation laws. *at-Automatisierungstechnik*, 308–322.
- Quintard, M., Kaviany, M., and Whitaker, S. (1997). Two-medium treatment of heat transfer in porous media numerical results for effective properties. *Advances in Water Resources*.
- Scheuermann, T.M., Kotyczka, P., Zanota, M.L., Pitault, I., Louati, H., and Maschke, B. (2019). Numerical approximation of heat transfer on heterogenous media. *PAMM*, 19(1). doi: 10.1002/pamm.201900372.
- Seslija, M., Scherpen, J.M.A., and van der Schaft, A.J. (2014). Explicit simplicial discretization of distributed-parameter port-Hamiltonian systems. *Automatica*, 50(2), 369–377. doi: 10.1016/j.automatica.2013.11.020.
- Tonti, E. (2001). A direct discrete formulation of field laws: the cell method. *Computer Modeling in Engineering and Sciences*, 2(2), 237–258. doi:10.3970/cmesc.2001.002.237.
- van der Schaft, A.J. and Maschke, B. (2013). Port-Hamiltonian systems on graphs. *SIAM Journal of Control and Optimization*, 51(2), 906–937.