STAG: Smart Tools and Applications in Graphics (2018)
M. Livesu, G. Pintore and A. Signoroni (Editors)

Direct Scalar Field - to - Truss Representation and Stress Simulation of Open Pore Domains

J. M. Munoz[1], O. Ruiz-Salguero[1], D. Montoya-Zapata[1,2], C. Cortes[3] and C. Cadavid[4]

[1]Laboratory of CAD CAM CAE, Universidad EAFIT, Colombia
[2]Industry and Advanced Manufacturing, Vicomtech, Spain
[3]eHealth and Biomedical Applications, Vicomtech, Spain
[4]Functional Analysis Group, Universidad EAFIT, Colombia

Abstract

In the domain of lattice and porous material geometric modeling, the problem of data size is central. When using full 3D manifold Boundary Representations (BRep), even the smallest domains engender staggering amounts of 3D finite elements. A partial solution has been implemented, which represents slender solid neighborhoods with non-manifold Boolean union of 1-manifolds (curves) and/or 2-manifolds (surfaces), added with thickness information, called 1.5D and 2.5D models, respectively. Automatic applications of these techniques requires the estimation of the medial axis of the porous media, to produce a truss or frame FEA. Previous works require explicit synthesis of the skin of the porous domain. This manuscript presents an alternative in which the medial axis and thus the 1.5D (truss) representation of the porous domain is directly obtained from the scalar field (i.e., Computer Tomography -CT-) of the domain, thus avoiding the explicit calculation of the domain skin. The manuscript also presents the noise removal and linearization of the medial axis data, to obtain the skeleton truss graph (including bar radii), that represents the porous domain. Shear and tension load simulations are conducted with the Truss model, showing that the generated model can be used in FEA software. Future work is required in extending this concept to lattice materials, where the medial axis includes surfaces and not only curves, as in this manuscript.

1. Introduction

Porous and lattice materials (e.g. foams) are particularly important in aerospace, medicine, additive manufacturing, etc. The geometric modeling of such materials presents intractably large data sets. This circumstance, in turn, hinders computations, behavior assessments, and design. In the particular domain of open pore materials, it is possible to use a Truss or Frame modeling (called 1.5D modeling) to study reasonable sized domains, which would be expensive to model and simulate, in terms of human manual work time and computational cost, by using full Boundary Representation (3D BReps).

Computer Tomography (CT) is a common method to sample foam or reticular materials. CTs basically produce a discrete scalar function $f : \mathbb{R}^3 \rightarrow \mathbb{R}$ that characterize the space occupancy of the open pore material. In CTs, f is presented in form of voxel (Volumetric Pixel) sets. Previous mechanical truss modeling of open pore materials ([COU*18]) takes CT as input, and explicitly produces the iso-surface $\partial B\ f(p) = c$ which bounds the solid material B. Then, the skeleton of ∂B ($SK(B)$) is computed. Finally, $SK(B)$ is used to define the Truss graph, which is the input to FEA methods to compute stress - strain models of B.

The present manuscript presents a method to avoid the explicit calculation of the skin ∂B to compute the skeleton $SK(B)$. Our method estimates $SK(B)$ directly from the scalar field f. An intermediate step is to estimate the Medial Axis of B, $MA(B)$. This medial axis $MA(B)$ presents a number of degeneracies which must be corrected before $SK(B)$ can be estimated in the form of a truss graph and used to model the material. This manuscript presents such a cleaning process.

This manuscript also presents an application of the direct f - to - $SK(B)$ skeleton extraction, in the particular area of strain - stress computation. The Truss model of an open pore material is then used for a FEA mechanical simulation, showing that the simplified truss representation obtained from the scalar field is a viable alternative to 3D Brep - based FE models.

Fig. 1(a) shows a portion of a porous material which in a large extent accepts a characterization in the form of a truss or frame, built with fastened struts or limbs.

Fig. 1(b) conveys the fact that the limbs have skeletons, which are curves c_i and have a local radius at each point of the curves. The curves meet in general in non-manifold nodes, as more than 2 curves are incident to them. Nodes with n curves incident to them are loci of the *Star-n* (*Sn*) type.

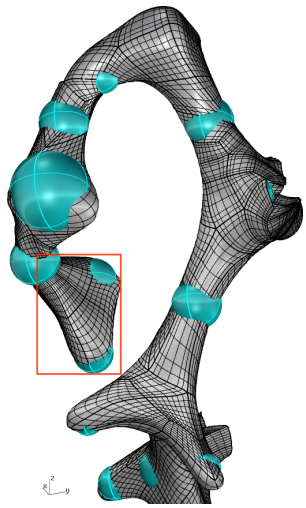
However, this example contains a non-processed subset (marked

DOI: 10.2312/stag.20181295

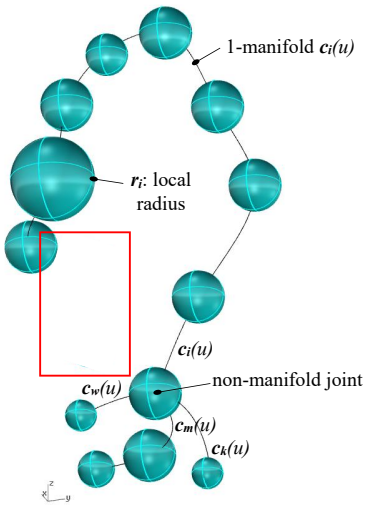
with a rectangular window), whose skeleton is a surface, and not a curve. In this manuscript, we assume that the foam material contains no such wall - like portions. A formalization of this discussion follows.

k-manifold. A set M in $\mathbb{R}^n$ is said to be k-manifold if for every point $p \in M$, there exists $\delta > 0$ such that for all radius $0 < r < \delta$, the set $M \cap Ball(p,r)$ is homeomorphic to an open disk in $\mathbb{R}^k$ ([Spi71]), where $Ball(p,r)$ is an open ball centered at p with radius r. Informally speaking, 1-manifolds are curves and 2-manifolds are surfaces, with no self-intersections in both cases.

Open Pore Materials. These structures contain spaces empty of material, or cavities, which are all connected.



(a) Point set Skin ∂B and Tangent Balls.



(b) Skeleton from Solid in Fig. 1(a). Limbs are formed by curves (e.g. $c_i(u), c_w(u), 0 \le u \le 1$) and local radius r_i.

Figure 1: *Porous Material with mostly 1D skeleton. Materials containing a 2D skeleton (see rectangular window) are not in the scope of this work.*

Medial Axis. Given a 2-manifold smooth closed surface ∂B which bounds a solid B, the medial axis of ∂B, $MA(B)$ (Fig. 1(b)), is the set of all voxels v that are centers of balls $Ball(v,r)$, centered in v with some radius $r > 0$ such that each ball $Ball(v,r)$ is tangent to ∂B in exactly 4 points.

Skeleton. Is the graph of voxel centers with the same connectivity as $MA(B)$. It may be considered as the wire version of $MA(B)$. Each point of $SK(B)$ contains the mass transferred to it by the mass conservation strategy in the iterative thinning of B.

Truss Graph. The bar + node representation of $SK(B)$, with the radii and dimensions of bars and nodes being determined by the mass information present in $SK(B)$. The Truss graph includes kinematic and torque restrictions required to mimic joints of the physical equivalent frame.

Star-n sets in $\mathbb{R}^2$. A star-n (Fig. 2) in this manuscript refers to an open set of points $p \in \mathbb{R}^2$, which is the Boolean union of $n = 2,3,4,...$ straight line segments. Each segment contains the origin in one end, and is open in the other end (Fig. 3).

Bar Pore Material. A *bar pore material* B is the set of points $p \in \mathbb{R}^3$ whose medial axis $MA(B)$ is composed by either (a) finite points in B whose neighborhood in B is homeomorphic to a star Sn with $n > 2$, or (b) infinite points in B, whose neighborhood in B is homeomorphic to $S2$.

The point set B whose boundary ∂B is shown in Fig. 1(a) is not a bar pore material because the portion in the rectangular window has medial axis which is 2-manifold (surface) and not 1-manifold (curve).

This manuscript refers to Bar Pore Materials and it is organized as follows: Section 2 reviews the relevant State of the Art. Section 3 explains the applied methodology. Section 4 presents the results of the implementation used and its application. Section 5 concludes the manuscript and suggests domains for future work.

2. Literature Review

First, a recent survey on the state of the art in skeletonization by [TDS*16] is discussed. Then, focus is made on (a) contraction and (b) thinning methods. Finally, skeletonization of foams is discussed.

2.1. Skeletonization algorithms

Skeletonization provides an effective and compact representation of an object by reducing its dimensionality to a *medial axis* or *skeleton* while preserving the topological and geometrical properties of the object [SBS16]. For a 3D solid, a skeleton could be a surface skeleton (2D) or a curve skeleton (1D). We are interested in objects with 1D skeletons because they allow simplification and acceleration [TDS*16] of the FEA problems.

The most common methods for computing analytical curve skeletons are: (a) Medial Surface - based, (b) Contraction methods, and (c) mesh decimation methods. The Medial Surface - based methods consist of identifying curves on the surface skeletons.

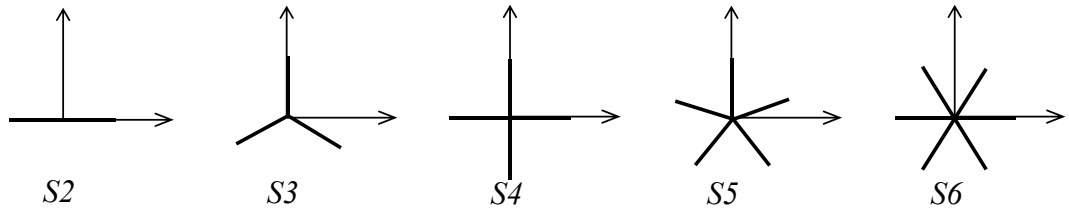


Figure 2: *Open point sets Star-n, S2, S3, S4, S5, S6 in $\mathbb{R}^2$. Note: Sn sets are open as they exclude the branch ends. The term open set star has no relation with open pore.*

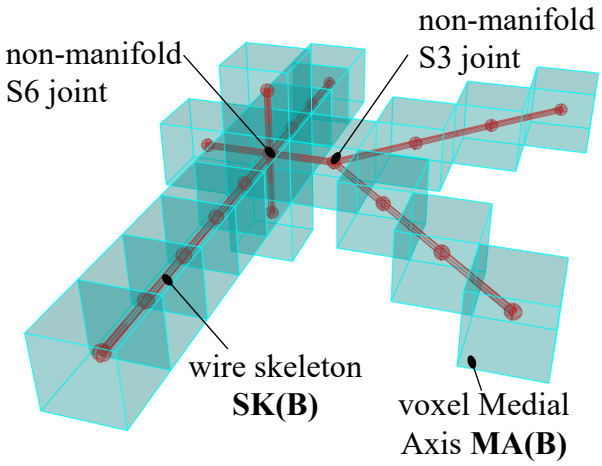


Figure 3: *Voxel Medial Axis MA(B), wire skeleton SK(B), S3 and S6 non-manifold star joints.*

Such methods are highly expensive [TDS*16] as they require accurate computation of geodesics between all points in a surface skeleton [DS06].

Contraction methods establish a shape evolution from the shape boundary to the curve skeleton [YT08]. Another contraction method is the discretization of the mean curvature flow in differential geometry. This method traces the surface-area loss through time steps [RW03].

Mesh decimation methods use iterative edge collapse on a triangle mesh [LWTH01] to converge towards the skeleton. However, the result is not necessarily smooth or centered, according to [TDS*16].

Curve skeletons from CT. For Computer Tomography scalar fields, contraction methods are directly applicable since they work on voxels. Refs. [Pud98,PK99] present algorithms where voxels are removed from the object boundary (skin) while preserving connectivity. A method to classify voxels that can be removed is described in [MB92, BM94]. It must be noticed that, in contraction methods, the order of voxel removal (thinning order) severely affects the final result [JST16]. Ref. [SBTZ02] presents a divergence - driven thinning that uses a sorted heap to ensure the correct processing order. Ref. [JST16] uses a voxel density order to prevent the jagging in the curve medial axes. This reference also presents a scalar attribute of the surviving voxels. At each evolution step, the scalar represents the mass of the voxels that have been removed, being this mass transferred to the surviving voxels. However, the mass of the whole domain cannot be retrieved from the final skeleton.

2.2. Models from Porous Materials

The generation of geometric models of foam microstructure and properties can be classified into (a) statistically generated models and (b) models from physical samples, as in [COU*18].

Statistically generated models. They characterize complex solid geometry by using its morphological parameters [BAM*17]. The approaches generate typical structures, with statistical variations, spanning a large domain by using pre - defined local arrangements. Used methods are: (a) arrays of identical cells (e.g. kelvin cells [DTH*11]) or (b) stochastic approaches such as Voronoi tessellations, ellipsoid overlapping, etc [JKK10]. In any case, all methods aim to generate a geometry and topology that mimics typical physical specimens. It is difficult for these methods to grasp the large variation in cell sizes and shapes [Lau08] and manufacturing defects of physical samples. If 3D full Breps are generated, the 3D meshing and simulation processes may not converge due to factors such as data size, data quality, and even software license constraints (as reported in [COU*18]). In this case, the relevance of the present manuscript (direct skeletonization from CT data) is evident.

Models from Physical Specimens. Physical specimens from foams, porous or reticular materials are usually sampled using micro-computed tomography (mCT). The 3D scalar field of such scans is expressed in voxel arrays with resolutions around 0.5 microns and domain sizes in the cm range [WS13]. Full 3D modeling from these samples may be conducted by either keeping the voxel set B, or computing a smooth boundary (skin) of B, ∂B. FEA simulations and visualizations are then conducted [MSOT10, SUJ*12, RNM15, NCSR*15]. The full 3D modeling is very precise and faithful to the physical sample. However, it is intractable for even the smallest cm-range samples, due to the staggering amount of data and labor required. Computing of smooth BReps from scalar fields is typically achieved by variations of the Marching Cubes algorithm [LC87], followed by intensive human labor in correcting the violations to 2-manifold properties that are common in triangle meshing. After the BRep is obtained, the FEA 3D meshing and computing may not converge even with small domains.

Ref. [COU*18] uses as input a correct BRep, ∂B of the porous

Approach	Refs.	Advantages	Disadvantages
Foam models from physical specimens.	[COU*18] [SUJ*12] [RNM15] [NCSR*15] [MSOT10]	- Preservation of geometry and topology of the original sample. - Efficient computing of stress - strain models of foams using Truss graphs [COU*18].	- Computing- and labor- expensive Breps required. - Expensive Full 3D FEA required.
3D characterization of simple points for thinning algorithms.	[MB92] [BM94]	- Calculation of the number of holes or genus not required.	- The proposition given to classify a simple voxel is correct, but the proposed pseudo-code in Ref. [MB92] may misclassify a simple voxel v as non-simple.
Volume contraction using thinning orderings.	[JST16] [SBTZ02]	- Thinning orderings for diverse applications. - [JST16] keeps track of the mass of voxels in the surface. - Surface or curve skeleton available through additional computation.	- Portion of initial mass of B missing in skeletonization [JST16]. - Disconnection in the curve skeleton topology [JST16]. - Boundary of the contracting shape not explicitly available in Ref. [SBTZ02].

Table 1: *Literature Review summary.*

domain. It produces a Truss graph representation of the strut frame implicit in ∂B. The Truss representation is then analyzed with finite elements, taking advantage of the so called *1.5D* finite elements, which are 1-dimensional ones equipped with local diameter (i.e. rods). This work shows important advantages from the Truss (i.e. 1.5D) over the full 3D models, showing that the truss model is less computationally expensive to simulate and that it can render reasonable strain - stress results. The method also has shown a good preservation of the porosity (ratio of empty to total volume) of the real sample with the Truss model.

2.3. Conclusions of the Literature Review

The main advantages and disadvantages of the surveyed methods are summarized in Table 1.

The contributions of this manuscript, with respect to the existing methods are:

1. Synthesis of the Truss graph directly from the Voxel scalar field B, without passing by the computing- and labor - expensive skin ∂B. Notice that this is an important advantage with respect to [COU*18], where a smooth, watertight, manifold, high-quality triangular mesh is required to obtain the medial-axis using the Mean Curvature Flow approach reported in [TAOZ12].
2. To achieve (1), we use and improve on [JST16] in the following aspects:
   a. Our algorithm keeps precise track of the mass of the initial solid B. Each voxel of the final Medial Axis $MA(B)$ accounts for representing a quantified contribution to keep the initial mass. This feature permits a precise estimation of the local radii along the struts of the Truss graph.
   b. Our algorithm conserves hanging branches of B for the purpose of representing their medial axis.
   c. Our algorithm avoids possible disconnections in the curve skeletons that may occur in [JST16].
3. We illustrate an application of the synthesized Truss graph in mechanical (stress - strain) computations.

3. Methodology

In this section we present the implemented method to estimate a truss graph simplification from the skeleton $SK(B)$. We use as input the CT of the sample, expressed as a scalar field f in the form of voxels. In Section 3.1, we calculate the medial axis of B ($MA(B)$), which contains information about the mass of B. Then, in Section 3.2, we use $MA(B)$ with the mass information to find a truss graph simplification of B, which is well suited to model bar pore materials.

3.1. Medial Axis from Scalar Field Extraction

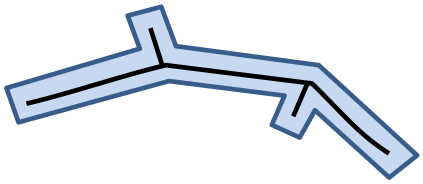
Algorithm 1 displays the main features of the extraction of the voxel Medial Axis from the domain B (voxels from the CT scan). Our algorithm is inspired in the boundary density transport approach of [JST16].

Algorithm 1 Extraction of Medial Axis from Scalar Field (CT).

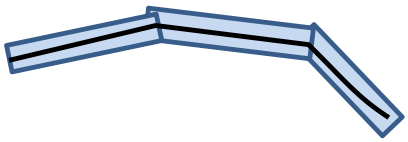
```
1:  procedure MA = FIELD_TO_MEDIAL_AXIS(B : voxel set)
2:      ∂B = boundary(B)
3:      superfluous = find_simple( ∂B )
4:      while superfluous ≠ [ ] do
5:          v = first( superfluous )
6:          n = inner_normal( ∂B, v )
7:          N(v) = neighbors( ∂B, v, n )
8:          mass( N(v) ) = mass( N(v) ) + mass(v)
9:          superfluous = superfluous − {v}
10:         ∂B = update( ∂B, v )
11:         superfluous = update(superfluous, ∂B, v)
12:     end while
13:     MA = ∂B
14: end procedure
```

In the line 3, the `superfluous` set is chosen, which is composed of voxels in the boundary ∂B which (a) do not change the voxel set connectivity when missing (i.e. *simple voxels* [MB92, BM94]), and (b) participate of a 1-manifold violation. We add the

condition (b) to avoid that the medial axes of hanging branches disappear. Voxels which do not change connectivity but belong to a hanging branch are not `superfluous`. This is an important difference with previous methods (e.g. [JST16]), where 1-manifold wires are considered only if they have received a large amount of mass from the boundary voxels. Since, by definition, every voxel of a hanging or dead-end branch does not change connectivity of the set, using only connectivity as elimination criterion would lead to the whole dead-end branch to be eliminated. Condition (b) above forces to eliminate only voxels of such branches which additionally violate 1-manifoldness. Line 4 indicates that the iteration stops when only essential voxels remain in B. Line 5 chooses for elimination a voxel in the `superfluous` set which has a minimal value of accumulated mass in it. Line 6 computes the inner pointing vector normal n to the skin ∂B at v using a gradient estimation (Neumann et al [NCKG00]). Line 7 identifies the 26-neighborhood N_{26} [MB92] associated with the voxel v to delete.

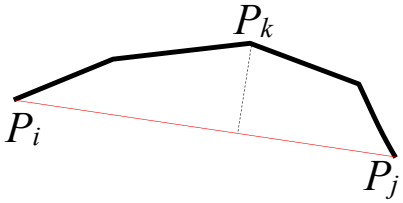


(a) Medial Axis with Noise (hairs).

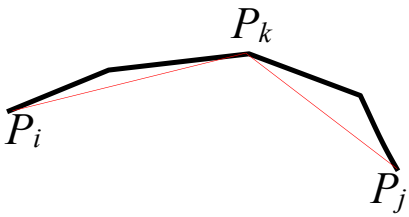


(b) Medial Axis without Noise.

Figure 4: *Hair removal from the Medial Axis $MA(B)$.*



(a) Large Linearization Error.

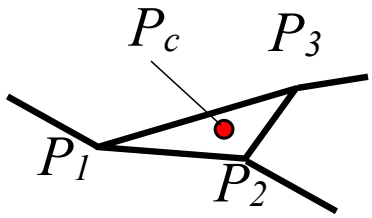


(b) Small Linearization Error.

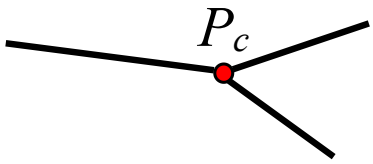
Figure 5: *Recursive subdivision to obtain a small linearization error from $MA(B)$.*

Line 8 transfers the mass of voxel v to its neighbors in direction n, using diffusive advection (Jalba et al. [JST16]). In this approach, the major receptor of mass is the voxel in the neighborhood of v that occupies the n direction. The other voxels in the neighborhood of v also receive minor proportions of the v mass. At this time the mass associated to v is null (fully transferred to its neighbors). In lines 9 and 10, voxel v is eliminated from the `superfluous`, B and ∂B sets. Now, the boundary ∂B is updated around v. Line 11 recomputes the B connectivity since the absence of v changes it. Thus, the `superfluous` voxel set must be updated.

When the `superfluous` voxel set is empty, $B = \partial B = MA(B)$. Line 13 recognizes this fact.



(a) 3 - cycle in Medial Axis.



(b) 3 - cycle eliminated from Medial Axis.

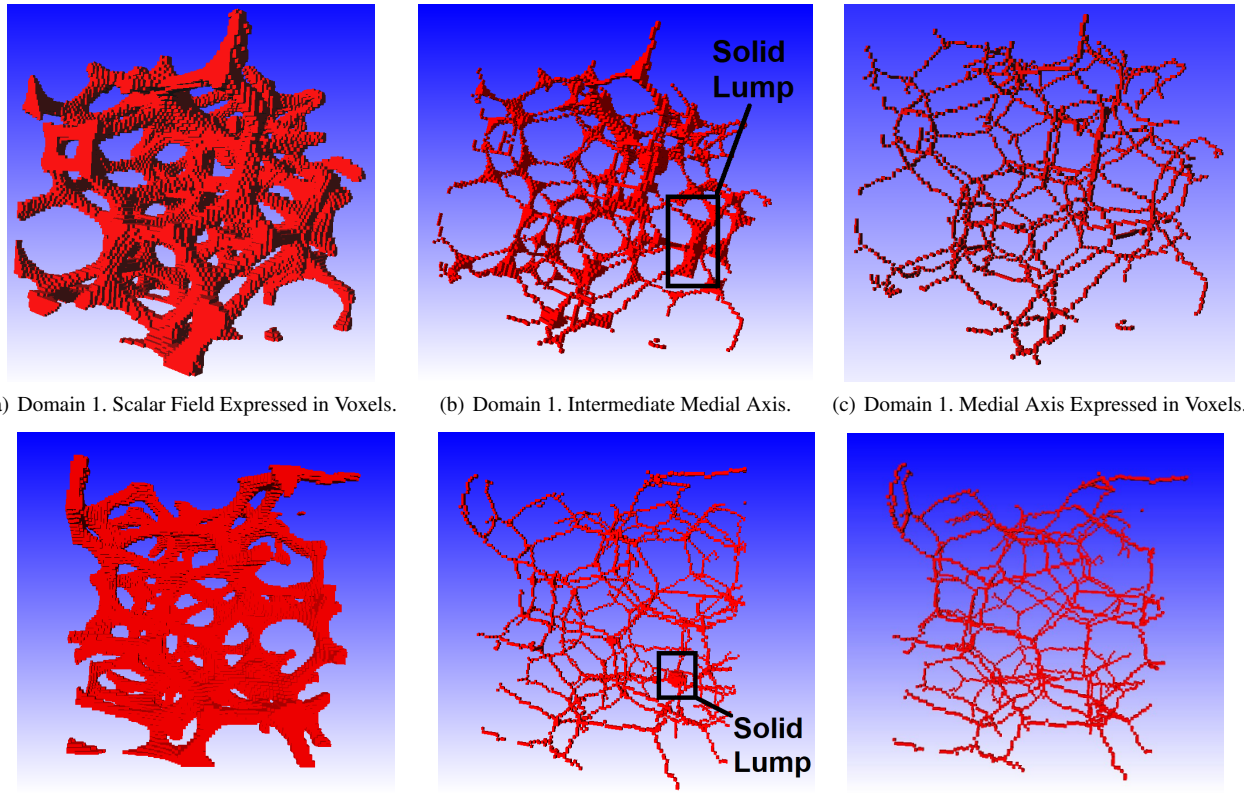
Figure 6: *Elimination of 3-cycles from $MA(B)$.*

Algorithm 2 Conversion from the Medial Axis $MA(B)$ into the Truss Graph (V,E)

```
1: procedure [V, E] = MEDIAL_AXIS_TO_TRUSS( MA : Medial Axis Graph )
2:     MA = hair_removal( MA )
3:     MA = edge_linearization( MA )
4:     SK = cycles_removal( MA )
5:     [V, E] = radii_estimation( SK )
6: end procedure
```

3.2. Truss Graph from Medial Axis

The Medial Axis $MA(B)$ obtained from the thinning algorithm in Algorithm 1 contains several characteristics, which make it useless as skeleton for a Truss graph (Fig. 8(a)). These features are: (1) high level of noise that manifests in small dead-end paths (hairs), (2) oscillations and high curvatures in the node - to - node paths, and (3) irreducible cycles of 3 edges (3-cycles). Algorithm 2 presents a description of the procedure to obtain the Truss graph from the Medial Axis. Steps 2-4 outline the cleaning of $MA(B)$ by removing features (1)-(3). Then, the mass information is used to produce the truss graph.



(a) Domain 1. Scalar Field Expressed in Voxels. (b) Domain 1. Intermediate Medial Axis. (c) Domain 1. Medial Axis Expressed in Voxels.

(d) Domain 2. Scalar Field Expressed in Voxels. (e) Domain 2. Intermediate Medial Axis. (f) Domain 2. Medial Axis Expressed in Voxels.

Figure 7: *Voxel - based Scalar Field and its Medial Axis. Data Sets 1 and 2*

To advance towards a Truss graph, noise in the form of hairs must be removed (Fig. 4). The short excursions or short dead-end paths are removed. An excursion is classified as *short* if its length is smaller than a given parameter L. In this work, L is equivalent to the biggest value of the Transform of Euclidean Distance [MRQ01] from $MA(B)$ to the original skin of B (∂B).

Once the hairs have been suppressed, it is necessary to execute a linearization of the resultant medial axis to eliminate oscillations and high curvatures in the node - to - node paths. The high level of noise in Fig. 5(a) is removed by replacing quasi linear paths with a straight edge. If the path deviates in significant manner (Fig. 5) from a straight one, a recursive subdivision is conducted until each linear edge does not deviate more than a threshold P from its curved equivalent. We have taken $P = 0.2$, which means that the polyline can deviate at most 20% from the straight line.

Irreducible cycles of 3 edges (3-cycles) appear in the wireframe Medial Axis $MA(B)$ (Fig. 6). They are removed by replacing vertex points p_1, p_2 and p_3 in Fig. 6(a) by the barycenter of the triangle p_c in all relevant graph topology. After the 3-cycles are suppressed, the skeleton $SK(B)$ is obtained.

Wire connectivity in $SK(B)$ is used in conjunction with the mass information to produce a truss graph $[V,E]$ (e.g. Figs. 8(c), 8(g)). Elements in V and E are assumed to be spheres and cylinders, respectively. Finally, we estimate the radius of each element in V and E by using its geometric and mass data, as shown in step 5 of algorithm 2.

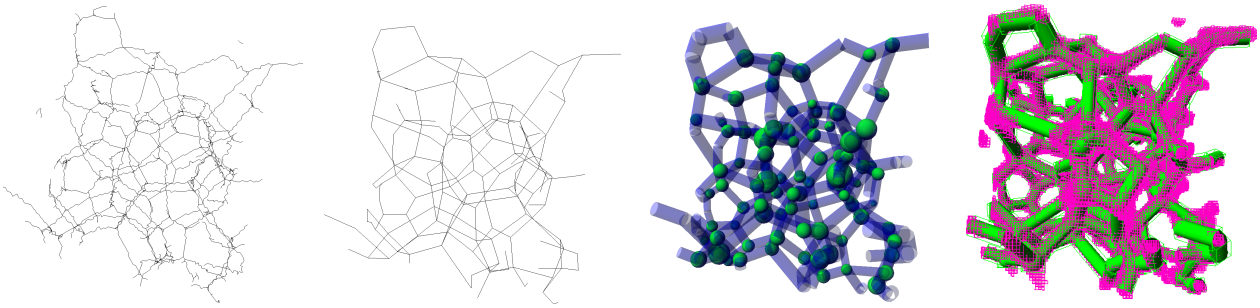
4. Results

Section 4.1 shows the result of the application of Algorithms 1 and 2 to two different samples, and gives a comparison of the porosity (a characteristic parameter for porous media) between the Truss graph and the initial shape. Section 4.2 shows an example in which the Truss graph approximation can be used to generate relevant mechanical simulations.

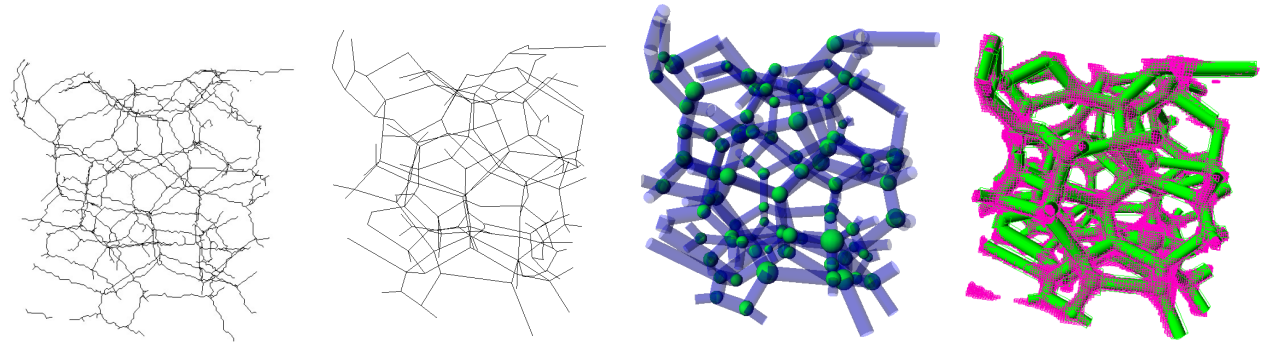
4.1. Truss Representation of Voxel Sets

Fig. 7 presents an initial voxel - based scalar field and an intermediate result of the Medial Axis extraction (Algorithm 1). It is relatively simple to detect that the thinning has not finished since Figs. 7(b) and 7(e) show solid lumps of voxels still present. These lumps are in violation of 1-manifold or *Sn* (Fig. 2) conditions. Figs. 7(c) and 7(f) show the final medial axis representation, which in all neighborhoods presents 1-manifold or *Sn* conditions.

Fig. 8 presents aspects of the processing of a medial axis $MA(B)$ to convert it into a usable skeleton $SK(B)$ and then into a Truss graph representation of B (Algorithm 2). The medial axis is usually very noisy (Fig. 8(a)), containing hairs, irreducible triangular



(a) Domain 1. Wireframe Noisy Skeleton. (b) Domain 1. Wireframe Clean Linearized Skeleton. (c) Domain 1. Truss Graph Approximation. (d) Domain 1. Voxel Set and its Truss Graph Approximation.



(e) Domain 2. Wireframe Noisy Skeleton. (f) Domain 2. Wireframe Clean Linearized Skeleton. (g) Domain 2. Truss Graph Approximation. (h) Domain 2. Voxel Set and its Truss Graph Approximation.

Figure 8: *Process of Approximation of a Voxel Set by a Bar / Sphere Graph. Data Sets 1 and 2*

cycles, and strongly curved struts. Figs. 4, 5 and 6 intuitively explain the hair removal, recursive linearization and 3-cycle elimination, respectively. The result is a clean linearized skeleton $SK(B)$, as presented in Fig. 8(b).

Fig. 8(c) presents the equivalent Truss Graph (V,E). In this graph, the nodes in V contain information about their (x,y,z) position in $\mathbb{R}^3$ and the equivalent radius of an envelope sphere that represents the material accumulated in the *Sn* joints. The edges in E contain information of their end vertices as well as the equivalent cylinder radius, as per the accumulated mass $m(v)$ of the surviving voxels in $MA(B)$. The radii associated with the *Sn* joints are not used in this manuscript, but are relevant to the void vs. full space ratio (i.e. porosity) of the material. Fig. 8(d) contrasts the initial (voxel) scalar field against its Truss graph approximation.

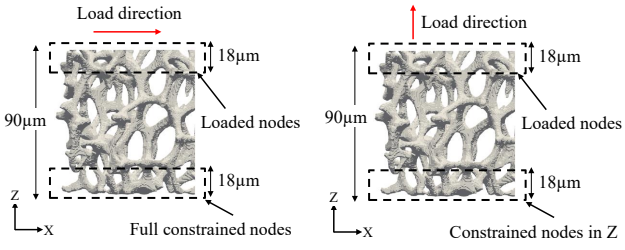
In order to get a measure of the quality of the truss graph approximation, the porosity (or void fraction) of the initial samples is compared with the porosity of their respective truss approximations (Table 2). The reader may notice that, for both samples, the difference of the porosity between the actual and the truss approximated domain is lower than 1%.

Domain	Actual porosity	Truss graph's porosity	Relative error
Domain 1	93.90%	94.01%	0.12%
Domain 2	93.32%	93.30%	0.02%

Table 2: *Comparison of the Porosity of the Actual Samples vs. the Porosity of the Truss Approximation.*

4.2. Application on Open Pore Stress - Strain Calculation

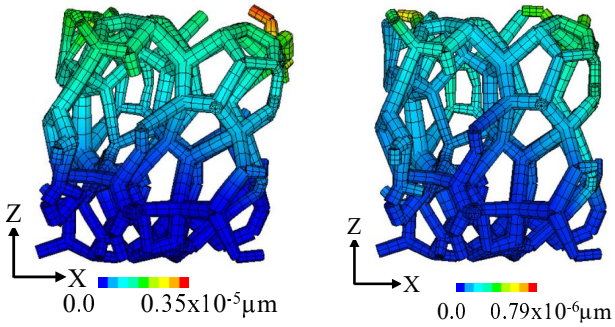
This section presents a proof - of - concept in the sense of showing that the Truss graph approximation of a voxel scalar field is indeed advantageous for relevant mechanical analyses . The particular example domain chosen is the stress - strain behavior of the porous material, as computed with Finite Element Analysis (FEA).



(a) Shear Test Conditions. (b) Tension Test Conditions.

Figure 9: *Boundary Conditions for the Mechanical Tests.*

In order to gain some insights on the computational savings of

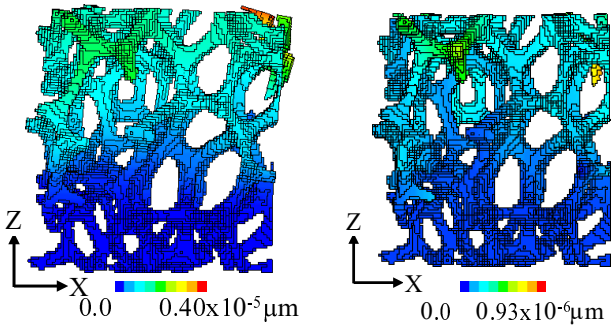


(a) Shear Deformation in X direction.

(b) Tension Deformation in Z direction.

Figure 10: *Shear and Tension FE Analyses with Truss model. Displacement scales are augmented by the FEA software for visualization.*

the Truss model against other traditional models, we have analyzed the Porous Domain 1 by generating two case studies of shear and tension loads (Fig. 9) for the corresponding (a) 3D voxel - based model and (b) Truss - based model. The 3D voxel - based model is generated by converting every Voxel into a hexahedral (cubic) element. Despite this technique does not represent the geometry of the domain with the same fidelity as a BRep model, it has been used for multiple studies of the mechanical response (estimation of apparent mechanical properties) of porous materials ([CDB17, SAKS05]).



(a) Shear Deformation in X direction.

(b) Tension Deformation in Z direction.

Figure 11: *Shear and Tension FE Analyses with the Voxel Model. Displacement scales are augmented by the FEA software for visualization.*

The results of the simulations are shown in Figs. 10 and 11. These figures are generated using a feature of the FEA software which allows to exaggerate the deformation for visualization purposes. Taking the maximum displacements of the Voxel model as reference values, the error of the Truss model in the estimation of X displacements in the shear test is 12.5%. In a similar fashion, the error in the estimation of Z displacements in the tension test is 15.1%.

Table 3 compares the FEA resources devoted to Voxel - based model vs. Truss - based simulations. The saving factor in all categories (FE nodes, elements, equations and memory) in favor of Truss graph data is in the order of 10^2. To determine how such saving factor impacts the computing time, we recur to the order of growth O of the execution time in terms of the number of nodes. In the worst-case scenario, the bandwidth of the stiffness matrix is $O(N^2)$, where N is the number of nodes. Hence, the time complexity of a FEA simulation is given by the term $O(N^3)$ ([FSSC11]). Then, a difference of 10^2 in the number of elements, implies a difference in the order of 10^6 in the number of operations that need to be performed to simulate the models.

Model	Number of Nodes	Number of Elements	Number of Equations	Memory allocated by Solver
Truss	729	786	3450	1.8MB
Voxel	95390	61039	230676	628.4MB

Table 3: *Computing Expenses for Truss vs. Voxel - based Foam Models.*

5. Conclusions

This manuscript presents a method to directly synthesize a truss graph representation from the scalar field (voxel CT) of an open pore material domain B. This approach is an alternative to previous ones which first require the estimation of the skin or Boundary representation ∂B. The implemented method enforces the conservation of volume information of each original domain bar, thus allowing to estimate the radii of the equivalent truss bars. The method implemented estimates the medial axis of the domain B, removes the noise (inherent to medial axes) and linearizes the local geometry. FEA computations are presented with the Truss model obtained from the scalar field, showing that it is a viable alternative to the 3D BRep - based models, which are much more expensive to generate and simulate.

Future work is required in porous or lattice domains whose medial axis includes surfaces (and not only curves). Additional work is needed in comparing the 1.5D Truss vs. 3D Brep simulations (considering the 3D one as ground truth). Also, experimental work is needed with actual porous material samples, to contrast the equivalent mechanical parameters estimated with the simulations vs. the laboratory values.

Acknowledgments

The authors thank the Institute of Mechanics (Prof. Holger Steeb), Ruhr-University Bochum (Bochum, Germany) who provided the data sets used in this work in 2012.

References

[BAM*17] BRACCONI M., AMBROSETTI M., MAESTRI M., GROPPI G., TRONCONI E.: A systematic procedure for the virtual reconstruction of open-cell foams. Chemical Engineering Journal 315 (may 2017), 608–620. URL: http://linkinghub.elsevier.com/retrieve/pii/S1385894717300700, doi:10.1016/j.cej.2017.01.069. 3

[BM94] BERTRAND G., MALANDAIN G.: A new characterization of three-dimensional simple points. Pattern Recognition Letters 15, 2 (feb 1994), 169–175. URL: http://linkinghub.elsevier.com/retrieve/pii/0167865594900469, doi:10.1016/0167-8655(94)90046-9. 3, 4

[CDB17] CHEN Y., DAS R., BATTLEY M.: Finite element analysis of the compressive and shear responses of structural foams using computed tomography. Composite Structures 159 (2017), 784 – 799. URL: http://www.sciencedirect.com/science/article/pii/S0263822316313605, doi:10.1016/j.compstruct.2016.09.091. 8

[COU*18] CORTES C., OSORNO M., URIBE D., STEEB H., RUIZ-SALGUERO O., BARANDIARAN I., FLOREZ J.: Geometry simplification of open-cell porous materials for elastic deformation fea. Engineering with Computers (2018), 1–20. URL: https://link.springer.com/article/10.1007/s00366-018-0597-3, doi:https://doi.org/10.1007/s00366-018-0597-3. 1, 3, 4

[DS06] DEY T. K., SUN J.: Defining and Computing Curve-skeletons with Medial Geodesic Function. Eurographics Symposium on Geometry Processing (2006). URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.92.5122{\&}rep=rep1{\&}type=pdf. 3

[DTH*11] DE JAEGER P., T'JOEN C., HUISSEUNE H., AMEEL B., DE PAEPE M.: An experimentally validated and parameterized periodic unit-cell reconstruction of open-cell foams. Journal of Applied Physics 109, 10 (may 2011), 103519. URL: http://aip.scitation.org/doi/10.1063/1.3587159, doi:10.1063/1.3587159. 3

[FSSC11] FARMAGA I., SHMIGELSKYI P., SPIEWAK P., CIUPINSKI L.: Evaluation of computational complexity of finite element analysis. In 2011 11th International Conference The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM) (Feb 2011), pp. 213–214. 8

[JKK10] JANG W.-Y., KYRIAKIDES S., KRAYNIK A. M.: On the compressive strength of open-cell metal foams with Kelvin and random cell structures. International Journal of Solids and Structures 47, 21 (oct 2010), 2872–2883. URL: http://linkinghub.elsevier.com/retrieve/pii/S0020768310002271, doi:10.1016/j.ijsolstr.2010.06.014. 3

[JST16] JALBA A. C., SOBIECKI A., TELEA A. C.: An Unified Multiscale Framework for Planar, Surface, and Curve Skeletonization. IEEE Transactions on Pattern Analysis and Machine Intelligence 38, 1 (2016), 30–45. doi:10.1109/TPAMI.2015.2414420. 3, 4, 5

[Lau08] LAUTENSACK C.: Fitting three-dimensional Laguerre tessellations to foam structures. Journal of Applied Statistics 35, 9 (sep 2008), 985–995. URL: https://www.tandfonline.com/doi/full/10.1080/02664760802188112, doi:10.1080/02664760802188112. 3

[LC87] LORENSEN W. E., CLINE H. E.: Marching Cubes: A high resolution 3d surface construction algorithm. SIGGRAPH Comput. Graph. 21, 4 (Aug. 1987), 163–169. URL: http://doi.acm.org/10.1145/37402.37422, doi:10.1145/37402.37422. 3

[LWTH01] LI X., WOON W., TAN T. S., HUANG Z.: Decomposing Polygon Meshes for Interactive Applications. Symp. on Interactive 3D Graphics (2001). URL: https://www.comp.nus.edu.sg/{~}tants/Paper/li-woon-tan-huang-01.pdf. 3

[MB92] MALANDAIN G., BERTRAND G.: Fast characterization of 3d simple points. In Proceedings., 11th IAPR International Conference on Pattern Recognition. Vol. III. Conference C: Image, Speech and Signal Analysis, (Aug 1992), vol. III, pp. 232–235. doi:10.1109/ICPR.1992.201968. 3, 4, 5

[MRQ01] MAURER C. R., RAGHAVAN V., QI R.: A linear time algorithm for computing the euclidean distance transform in arbitrary dimensions. In Information Processing in Medical Imaging (Berlin, Heidelberg, 2001), Insana M. F., Leahy R. M., (Eds.), Springer Berlin Heidelberg, pp. 358–364. 6

[MSOT10] MICHAILIDIS N., STERGIOUDI F., OMAR H., TSIPAS D.: An image-based reconstruction of the 3D geometry of an Al open-cell foam and FEM modeling of the material response. Mechanics of Materials 42, 2 (feb 2010), 142–147. URL: http://linkinghub.elsevier.com/retrieve/pii/S0167663609002014, doi:10.1016/j.mechmat.2009.10.006. 3, 4

[NCKG00] NEUMANN L., CSÉBFALVI B., KÖNIG A., GRÖLLER E.: Gradient Estimation in Volume Data using 4D Linear Regression. Computer Graphics Forum 19, 3 (sep 2000), 351–358. URL: http://doi.wiley.com/10.1111/1467-8659.00427, doi:10.1111/1467-8659.00427. 5

[NCSR*15] NATESAIYER K., CHAN C., SINHA-RAY S., SONG D., LIN C. L., MILLER J. D., GARBOCZI E. J., FORSTER A. M.: X-ray CT imaging and finite element computations of the elastic properties of a rigid organic foam compared to experimental measurements: insights into foam variability. Journal of Materials Science 50, 11 (jun 2015), 4012–4024. URL: http://link.springer.com/10.1007/s10853-015-8958-4, doi:10.1007/s10853-015-8958-4. 3, 4

[PK99] PALÁGYI K., KUBA A.: Directional 3D Thinning Using 8 Subiterations. Proc. Discrete Geometry Comput. Imagery (1999). 3

[Pud98] PUDNEY C.: Distance-Ordered Homotopic Thinning: A Skeletonization Algorithm for 3D Digital Images. Computer Vision and Image Understanding 72, 3 (dec 1998), 404–413. URL: http://linkinghub.elsevier.com/retrieve/pii/S1077314298906804, doi:10.1006/cviu.1998.0680. 3

[RNM15] RANUT P., NOBILE E., MANCINI L.: High resolution X-ray microtomography-based CFD simulation for the characterization of flow permeability and effective thermal conductivity of aluminum metal foams. Experimental Thermal and Fluid Science 67 (oct 2015), 30–36. URL: http://linkinghub.elsevier.com/retrieve/pii/S089417771400260X, doi:10.1016/j.expthermflusci.2014.10.018. 3, 4

[RW03] RUUTH S. J., WETTON B. T. R.: A Simple Scheme for Volume-Preserving Motion by Mean Curvature. Journal of Scientific Computing 19, 3 (2003). 3

[SAKS05] SAADATFAR M., ARNS C., KNACKSTEDT M., SENDEN T.: Mechanical and transport properties of polymeric foams derived from 3d images. Colloids and Surfaces A: Physicochemical and Engineering Aspects 263, 1 (2005), 284 – 289. A collection of papers presented at the 5th European Conference on Foams, Emulsions, and Applications, EUFOAM 2004, University of Marne-la-Vallee, Champs sur Marne (France), 5-8 July, 2004. URL: http://www.sciencedirect.com/science/article/pii/S0927775704010052, doi:10.1016/j.colsurfa.2004.12.040. 8

[SBS16] SAHA P. K., BORGEFORS G., SANNITI DI BAJA G.: A survey on skeletonization algorithms and their applications. Pattern Recognition Letters 76 (2016), 3–12. URL: http://dx.doi.org/10.1016/j.patrec.2015.04.006, doi:10.1016/j.patrec.2015.04.006. 2

[SBTZ02] SIDDIQI K., BOUIX S., TANNENBAUM A., ZUCKER S. W.: The Hamilton-Jacobi Skeleton. Int. J. Comput. Vis 48, 3 (2002), pp. 215–231. URL: http://www.cim.mcgill.ca/{~}shape/publications/iccv99.pdf. 3, 4

[Spi71] SPIVAK M. D.: Calculus On Manifolds: A Modern Approach To Classical Theorems Of Advanced Calculus. Westview Press, 1971. ISBN-13: 978-0-8053-9021-6, ISBN: 0-8053-9021-9. 2

[SUJ*12] SAENGER E., URIBE D., JÄNICKE R., RUIZ O., STEEB H.: Digital material laboratory: Wave propagation effects in open-cell aluminium foams. International Journal of Engineering Science 58 (sep 2012), 115–123. URL: http://linkinghub.elsevier.com/retrieve/pii/S0020722512000730, doi:10.1016/j.ijengsci.2012.03.030. 3, 4

[TAOZ12] TAGLIASACCHI A., ALHASHIM I., OLSON M., ZHANG H.: Mean curvature skeletons. In Computer Graphics Forum (2012), vol. 31, Wiley Online Library, pp. 1735–1744. 4

[TDS*16] TAGLIASACCHI A., DELAME T., SPAGNUOLO M., AMENTA N., TELEA A.: 3D Skeletons: A State-of-the-Art Report. Computer Graphics Forum 35, 2 (may 2016), 573–597. URL: http://doi.wiley.com/10.1111/cgf.12865, doi:10.1111/cgf.12865. 2, 3

[WS13] WILDENSCHILD D., SHEPPARD A. P.: X-ray imaging and analysis techniques for quantifying pore-scale structure and processes in subsurface porous medium systems. Advances in Water Resources 51 (jan 2013), 217–246. URL: http://linkinghub.elsevier.com/retrieve/pii/S0309170812002060, doi:10.1016/j.advwatres.2012.07.018. 3

[YT08] YU-SHUEN WANG, TONG-YEE LEE: Curve-Skeleton Extraction Using Iterative Least Squares Optimization. IEEE Transactions on Visualization and Computer Graphics 14, 4 (jul 2008), 926–936. URL: http://ieeexplore.ieee.org/document/4459323/, doi:10.1109/TVCG.2008.38. 3