Computers & Geosciences 101 (2017) 28–37

Contents lists available at ScienceDirect

Computers & Geosciences

journal homepage: www.elsevier.com/locate/cageo

Research paper

Accurate and efficient maximal ball algorithm for pore network extraction

Frederick Arand[a,b,*], Jürgen Hesser[b]

[a] Volume Graphics GmbH, Speyerer Str. 4-6, D-69115 Heidelberg, Germany
[b] Experimental Radiation Oncology, Medical Faculty Mannheim, Heidelberg University, Theodor-Kutzer-Ufer 1-3, D-68167 Mannheim, Germany

ARTICLE INFO

Keywords:
Pore-Throat-Network
MIS
Mathematical morphology

ABSTRACT

The maximal ball (MB) algorithm is a well established method for the morphological analysis of porous media. It extracts a network of pores and throats from volumetric data. This paper describes structural modifications to the algorithm, while the basic concepts are preserved. Substantial improvements to accuracy and efficiency are achieved as follows: First, all calculations are performed on a subvoxel accurate distance field, and no approximations to discretize balls are made. Second, data structures are simplified to keep memory usage low and improve algorithmic speed. Third, small and reasonable adjustments increase speed significantly. In volumes with high porosity, memory usage is improved compared to classic MB algorithms. Furthermore, processing is accelerated more than three times. Finally, the modified MB algorithm is verified by extracting several network properties from reference as well as real data sets. Runtimes are measured and compared to literature.

1. Introduction

Pore Networks are used in many fields to model porous media, for example in the evaluation of cellular materials (Knackstedt et al., 2006; Fischer et al., 2009; Benouali et al., 2005; Viot et al., 2008) and in the examination of rocks and sands (Andrä et al., 2013; Wildenschild and Sheppard, 2013; Hormann et al., 2016; Homberg et al., 2014; Al-Kharusi and Blunt, 2007; Dong and Blunt, 2009). All of the previous references contain a morphological analysis of the material under investigation, which underlines its importance, for some of them it is the central element of research.

In order to conduct future surveys in industry and academics, a robust algorithm, which performs automated morphology inspections on a wide range of materials, is desirable. Silin and Patzek (2006) introduced the method of Maximal Inscribed Spheres (MIS), which can be used to extract a network of pores and throats from a tomographic image. They argue that thinning algorithms, which are the basis of methods other than MIS, can lead to different results on the same data sets, in contrast to the robust MIS-based analysis. This reasoning together with the elegant concept of MIS have lead to the concentration and subsequent improvements of the method in this paper.

Rabbani et al. (2014) compared MIS to other morphological algorithms. Extensions and improvements to the original MIS-algorithm can be found in several contributions (Al-Kharusi and Blunt, 2007; Dong and Blunt, 2009; Byholm et al., 2006). Al-Kharusi and Blunt (2007) extracted complete pore networks along with pore, throat and coordination number statistics. However, due to processing time limitations, data sets were restricted to a relatively small size of 200^3 voxels. This is also noted by Dong and Blunt (2009), who improve previous works further.

Homberg et al. (2014) extracted pore networks using discrete Morse theory. Their idea of hierarchical pore merging is used in this work.

2. Methods

Basic concepts which are found in the first paper about the Maximal Ball (MB) algorithm (Silin and Patzek, 2006) and subsequent ones (Al-Kharusi and Blunt, 2007; Dong and Blunt, 2009; Byholm et al., 2006) are similar to those in this paper. However, methods, data structure and detailed concepts are changed fundamentally.

The group of existing papers will be referred to as *classic MB algorithms* cMBa, while the method presented here will be abbreviated as *modified MB algorithm* mMBa. They are compared step by step in this section. Many figures in this paper are in two dimensions (2D), however, they are also valid in 3D.

Coordinates $\mathbf{x}^i \in \mathbb{N}^3$ are called *voxel coordinates*. Volumetric data, e.g., reconstructed from computed tomography (CT) scanners, is stored as *voxel values* $f^i = f(\mathbf{x}^i)$, $f^i \in \mathbb{R}$. The set of voxel values is called *voxel volume* $\{f^i\}$, $i \in \{1,\ldots,N\}$, where N is the number of voxels in the volume. All following considerations assume monodisperse material, and that each f^i is proportional to the percentage of material contained

* Corresponding author at: Volume Graphics GmbH, Speyerer Str. 4-6, D-69115 Heidelberg, Germany.
E-mail address: fredi.arand@gmail.com (F. Arand).

http://dx.doi.org/10.1016/j.cageo.2017.01.004
Received 10 October 2016; Accepted 16 January 2017
Available online 24 January 2017

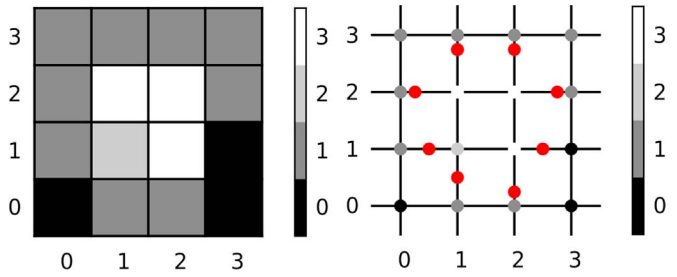

Fig. 1. *Left:* 2D image $\{f^i\}$ of size 4×4 with values ranging from 0 to 3. Pixels are depicted as tiles with different grey values, each tile represents a Voronoi cell. *Right:* A threshold $f_t = 1.5$ makes it possible to define boundary points (red) between void space (<1.5) and material (≥1.5). Pixels are depicted at grey valued dots at their actual coordinates. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

in the Voronoi cell induced by the regular grid $\{\mathbf{x}^i\}$, i.e., a *coverage representation* (Lindblad and Sladoje, 2015).

Noisy voxel data can produce artifacts such as false pores and throats during network extraction and should therefore be preprocessed. For example, datasets can be denoised with the well established non-local means filter (Buades et al., 2005; Darbon et al., 2008).

In order to extract a pore network from the volume, the latter is classified using two partitions: *void space* Ω_{void}, which contains *void voxels* $\mathbf{x}^i_{\text{void}}$, and *material* Ω_{mat}. Void voxels are defined as follows:

$$\mathbf{x}^i_{\text{void}} = \{\mathbf{x}^i \,|\, f(\mathbf{x}^i) < f_t\}$$

with the global *segmentation threshold* f_t.

Another preprocessing step can be applied in this context: One can require for the set $\{\mathbf{x}^i_{\text{mat}}\}$ to be interconnected via 26-neighborhoods and to the volume boundary $\partial\Omega$. All $\mathbf{x}^i_{\text{mat}}$ not satisfying this condition, i.e., being enclosed by void space, can be erased.

Both cMBa and mMBa process $\mathbf{x}^i_{\text{void}}$. For the mMBa, more geometry information is extracted from the voxel values using *grid value interpolation*, which is depicted in Fig. 1. *Grid line coordinates* (Lindblad and Sladoje, 2015) $\mathbf{x}_{\text{grid}}$ are defined on the edges that connect all grid vertices $\mathbf{x}^i$ and $\mathbf{x}^j$ in a 6-neighborhood. Each $\mathbf{x}_{\text{grid}}$ has a real value in one coordinate, the remaining two coordinates remain integers, e.g., $\mathbf{x}_{\text{grid}} \in \mathbb{R} \times \mathbb{N} \times \mathbb{N}$. *Grid line values* $f(\mathbf{x}_{\text{grid}})$ are defined by linear interpolation of direct neighbors f^i and f^j on grid lines. Now, *boundary points* $\mathbf{x}^k_{\text{bound}} = \{\mathbf{x}_{\text{grid}} \,|\, f(\mathbf{x}_{\text{grid}}) = f_t\}$ indicate the subvoxel accurate border between Ω_{mat} and Ω_{void} on grid edges of $\{\mathbf{x}^i\}$, similar to the marching cubes algorithm. The transformation of a coverage representation to a *grid line sampling* (Lindblad and Sladoje, 2015) with boundary points is used for the subsequent step of the mMBa, the calculation of a distance field.

2.1. Distance field

In the mMBa, a *distance field* $\{d^i\}$ with values $d^i = d(\mathbf{x}^i)$, $d^i \in \mathbb{R}^+$, is generated using the *Euclidean Distance Transform* (EDT) (Lindblad and Sladoje, 2015). Let $\{\mathbf{x}^k_{\text{bound}}\}$ be the set of boundary points, $k \in \{1, \ldots, K\}$, where K is the total number of boundary points found by grid line sampling. Then, the EDT determines $\{d^i\}$ as the shortest distance of each $\mathbf{x}^i$ to $\{\mathbf{x}^k_{\text{bound}}\}$:

$$d(\mathbf{x}^i) = \begin{cases} \min_k(\| \mathbf{x}^i - \mathbf{x}^k_{\text{bound}} \|_{22}) & \text{if } \mathbf{x}^i \in \Omega_{\text{void}} \\ 0 & \text{if } \mathbf{x}^i \in \Omega_{\text{mat}} \end{cases}$$

Voxel accurate distance transforms with linear time complexity are described by Meijster et al. (2000). The method was extended to subvoxel accuracy with respect to grid line sampled points by Lindblad and Sladoje (2015), which is used for the mMBa. An example for a distance field calculated from a coverage representation can be found in Fig. 2.

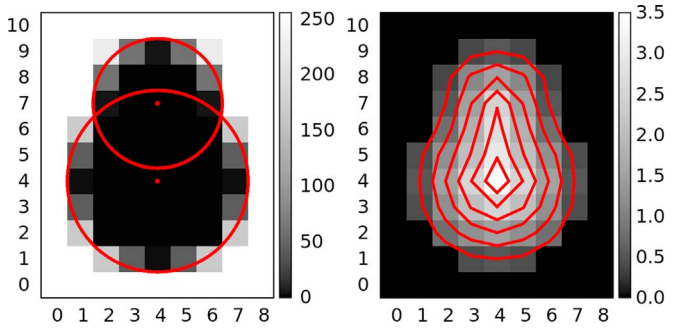

Fig. 2. *Left:* Two circles with center coordinates (4, 4), (4, 7) and radii 3.5, 2.5 are used to create a coverage representation: Scanning is simulated by taking 32×32 regular subpixel points for each pixel cell and testing if they are outside of both circles. Resulting hits are scaled to the range [0, 255]. *Right:* A distance field is created with $f_t = 127.5$. Contour lines (red) differ by 0.5 and range from 0.5 (outer line) to 3.0 (inner line). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

2.2. Maximal balls: Definition

Both cMBa and mMBa place *Maximal Balls* $\mathcal{MB}^i$ at every $\mathbf{x}^i_{\text{void}}$. By definition (Silin and Patzek, 2006), each $\mathcal{MB}^i$ is a sphere centered at $\mathbf{x}^i_{\text{void}}$, with radius r^i chosen maximal such that $\mathcal{MB}^i$ is completely contained in Ω_{void}.

In the cMBa, r^i takes discrete values and is obtained using inflating and deflating spheres (Dong and Blunt, 2009). $\mathcal{MB}^i$ are represented as voxelized, digital spheres (Dong and Blunt, 2009).

In the mMBa, for each $\mathcal{MB}^i$, $r^i = d^i$. This is equivalent according to the definitions of maximal balls and the distance field. The spectrum of radii from the distance field has a higher density than in the cMBa: Grid line sampling leads to a subvoxel resolution. In this sense, the mMBa generally is more accurate than the cMBa. Maximal Balls created with the mMBa are displayed in Fig. 3.

The accuracy gain in the order of one voxel is useful in two scenarios: First, small structures, i.e., which are up to few orders larger than 1 voxel, are characterized with greater detail. Second, binary data sets can be converted to grey value data together with the reduction of aliasing and terracing artifacts (Gibson, 1998). This step could be beneficial for the accuracy for the mMBa on binary data, but was not implemented in this paper.

2.3. Classic maximal ball algorithm

In the cMBa, after $\mathcal{MB}^i$ are assigned to $\mathbf{x}^i_{\text{void}}$, they are sorted in descending order by r^i. They are sequentially processed to build up a hierarchy (Silin and Patzek, 2006): For each $\mathcal{MB}^i$, all $\mathcal{MB}^j$ with $r^j \leq r^i$ and $\| \mathbf{x}^i - \mathbf{x}^j \|_2 \leq r^i$ become *children* of $\mathcal{MB}^i$, and $\mathcal{MB}^i$ becomes the *parent* of each $\mathcal{MB}^j$. Silin and Patzek (2006) used the terms *slave* and *master* instead. For each $\mathcal{MB}^i$, all parents are saved in a *parent list* $\mathcal{L}^i_{\text{parent}}$, and all children (irrespective of parents) are saved in a *child list* $\mathcal{L}^i_{\text{child}}$. A partial hierarchy graph is visualized in Fig. 3.

Next, the hierarchical graph is remapped to have two levels: If a parent does not have parents itself, it is identified as a *pore center* and all of its children's children are recursively added while replacing their previous parent by the pore center until no more children's children are found. Silin and Patzek (2006) describe this remapping algorithm of depth-first search type detail.

Finally, all $\mathcal{MB}^i$ are classified: If they have up to one parent, they are part of a *pore*. Individual pores are distinguished by their pore centers. If $\mathcal{MB}^i$ has two or more parents, it is defined to be part of a *throat* that connects the pores inside $\mathcal{L}^i_{\text{parent}}$. In Fig. 3, no throats are created and the network consists of one pore only.

Silin and Patzek (2006) introduced *voxel objects* $\mathcal{VO}^i$ and a *reference table* $\{rt^i\}$ to implement the concepts of the cMBa. They can be summarized as follows: