# Chapter 40
# Applications of Microtomography to Multiscale System Dynamics: Visualisation, Characterisation and High Performance Computation

**Jie Liu, Klaus Regenauer-Lieb, Chris Hines, Shuxia Zhang, Paul Bourke, Florian Fusseis and David A. Yuen**

**Abstract** We characterise microstructure over multiple spatial scales for different samples utilising a workflow that combines microtomography with computational analysis. High-resolution microtomographic data are acquired by desktop and synchrotron X-ray tomography. In some recent 4-dimensional experiments,

J. Liu (✉) · K. Regenauer-Lieb · F. Fusseis
Multi-scale Earth System Dynamics Group, School of Earth and Environment,
University of Western Australia, 35 Stirling Hwy, Crawley,
WA 6009, Australia
e-mail: liujieigcea@gmail.com

K. Regenauer-Lieb
Computational Geoscience, CSIRO ESRE, PO Box 1130, Bentley,
WA 6102, Australia

K. Regenauer-Lieb · F. Fusseis
Western Australian Geothermal Center of Excellence, PO Box 1130, Bentley,
WA 6102, Australia

C. Hines
iVEC, 'The hub of advanced computing in WA', PO Box 1130, Bentley,
WA 6102, Australia

S. Zhang
Supercomputing Institute, University of Minnesota, Minneapolis, MN 55455, USA

P. Bourke
iVEC @ University of Western Australia, 35 Stirling Hwy, Crawley,
WA 6009, Australia

D. A. Yuen
Department of Geology and Geophysics, University of Minnesota,
Minneapolis, MN 55455, USA

D. A. Yuen
Department of Earth Sciences and Minnesota Supercomputing Institute,
University of Minnesota, Minneapolis, MN 55455, USA

D. A. Yuen
School of Environmental Sciences, China University of Geosciences, Wuhan, China

microstructures that are evolving with time are produced and documented in situ. The microstructures in our materials are characterised by a numerical routine based on percolation theory. In a pre-processing step, the material of interest is segmented from the tomographic data. The analytical approach can be applied to any feature that can be segmented. We characterise a microstructure by its volume fraction, the specific surface area, the connectivity (percolation) and the anisotropy of the microstructure. Furthermore, properties such as permeability and elastic parameters can be calculated. By using the moving window method, scale-dependent properties are obtained and the size of representative volume element (RVE) is determined. The fractal dimension of particular microstructural configurations is estimated by relating the number of particular features to their normalized size. The critical exponent of correlation length can be derived from the probability of percolation of the microstructure. With these two independent parameters, all other critical exponents are determined leading to scaling laws for the specific microstructure. These are used to upscale the microstructural model and properties. Visualisation is one of the essential tools when performing characterisation. The high performance computations behind these characterisations include: (1) the Hoshen-Kopelman algorithm for labelling materials in large datasets; (2) the OpenMP parallelisation of the moving window method and the performance of stochastic analysis (up to $640^3$ voxels); (3) the MPI parallelisation of the moving window method and the performance of stochastic analysis, which enables the computation to be run on distributed memory machines and employ massive parallelism; (4) the parallelised MPI version of the Hoshen-Kopelman algorithm and the moving window method, which allows datasets of theoretically unlimited size to be analysed.

## 40.1 Introduction

Micro computed tomography (micro-CT or microtomography) is a technique developed from medical CT scans for detecting structures in the interior of objects with submicron resolution. It has three important characteristics: (1) providing digital information on 3-D geometries; (2) no destruction of the samples; and (3) offering high resolution. These advantages motivate a wide range of applications of microtomography for different materials, including rock, bone, ceramic, metal and soft tissue. When the resolution of microtomography reaches the nano-meter scale, it is also known as nano-CT or nanotomography.

3-D digital microtomographic data are comprised of a stack of greyscale images which can be loaded into volume visualisation software packages. Although micro-CT and visualisation have already provided much qualitative information of the interior of the object, quantitative analysis can provide additional insights. For example,

it can give answers to questions such as how large individual entities are and what are the relationships between different entities. In addition to the characterisation of individual entities, we are also interested in the statistical characteristics of the sample. From statistical analysis of the sample a representative volume element (RVE) can be determined. RVE's are statistically representative volumes containing a sufficiently large set of microstructure elements such that their influence on the average macroscopic property (porosity, elasticity, permeability, etc.) has converged when a larger volume element is taken. It is important to specify the size of RVE when estimating the physical properties based on a structured volume. Furthermore, some scaling parameters can be extracted from the microstructures and this makes it possible to upscale the properties that were computed at micro-scale. Thus, the quantitative analysis of microtomography is performed in three levels: characterisation of individual entities, statistical characteristics and determination of RVE size, and extraction of scaling parameters for upscaling.

A quantitative analysis requires high performance computing for three reasons. The first reason is the size of datasets. At present a general micro-CT dataset is $2048 \times 2048$ pixels per slice, and there are around 2048 slices, this corresponds to 8GB if every pixel is represented by 1 byte. Moving to higher resolutions, $4096^3$ volumes are now possible and it is expected that $8192^3$ volumes will be generated in the near future. Without high performance computing techniques it is difficult to deal with datasets of this scale simply because of the memory required to represent the data. The second issue is that the computation of statistical properties often requires significant processing power. Thirdly, CT scan experiments in a time series increase both the volume of data and the computational requirements linearly with the number of time steps.

Software does exist to perform quantitative microstructural analysis (Lindquist 2005; Nakashima and Kamiya 2007; Ketcham 2005). However, concerning the stochastic analysis, high performance computing, and extraction of scaling parameters, our strategy (Liu et al. 2009; Liu and Regenauer-Lieb 2010) is unique, more comprehensive, and more powerful. The aims of this research are (1) characterisation of microstructures by visualisation and quantitative analysis; and (2) upscaling of micro-scale properties by stochastic analysis and percolation theory. In this paper we emphasise the technical details and the computational methods employed.

## 40.2 Microtomographic Data

### 40.2.1 X-Ray CT Theory and Synchrotron CT

In micro-CT experiments, X-rays are created from a high-power source and directed at a sample. A detector on the opposite side of the sample measures the intensity of the transmitted X-rays. The sample is rotated step by step, generally in 1° steps, and the detector records an image for every step. The intensity of the X-rays reaching

the detector depends on the sample path length and the X-ray attenuation coefficient of the material that it passes though. The longer the path length and the greater the attenuation coefficient of the material, the greater the number of diffraction and scattering events, thereby weakening the X-ray signal reaching the detector. Using reconstruction algorithms (i.e. Fourier transform), the 2-D images are converted to a complete 3-D map of the sample.

The digital 3-D map of the sample is generally represented by a stack of slices, each a greyscale image. The brightness of pixels within the images is related to the atomic number, regions with higher atomic numbers will be brighter than regions with lower atomic numbers. Pores in the sample are always shown as the darkest regions and are the easiest to identify. Different minerals in rock samples are recognisable as long as there is a significant difference in their atomic number.

The principle of synchrotron CT is similar to the general X-ray CT, the difference lies in the energy source employed. The synchrotron uses the energy from a particle accelerator in which the magnetic field and the electric field are carefully synchronised with the travelling particle beam. Generally, synchrotron CT results in better image quality and fewer artifacts compared to an ordinary X-ray CT. The other feature of synchrotron CT is that it results in images with a higher dynamic range. For example, pixels in X-ray CT image are generally 8-bit quantities, whereas pixels in synchrotron CT image are 32-bit values. Higher dynamic range means regions with smaller differences in atomic number can be identified.

### 40.2.2 Pre-Processing of Data

Before quantitative analysis, the 3-D microtomographic data needs pre-processing to select and label the phase of interest. This target phase can be pores, solid grains or any kind of material, which can be identified within the volume by its intensity value. We use the Avizo® software package to load the images. By choosing a suitable threshold value, the greyscale images are converted to binary images where the black and white pixels represent different materials, e.g., pores and solids in porous media. This is referred to as the "segmentation procedure". The Avizo® software enables us to build up the 3-D binary model, to display the structures in pores or in solids, and to export the label-data. The label-data is a binary volumetric data file in which each voxel that has a value lower than the threshold is labelled as 0 and those greater than or equal to the threshold are labelled as 1. This label-data volume is used for all subsequent analysis.

Very often the sample is cylindrical but sometimes they are a more irregular shape. Our quantitative analysis requires a parallelepiped, thus it is necessary to crop a parallelepiped volume from the whole 3-D dataset. This additionally removes regions that contain non-material voxels. This procedure is also implemented within Avizo®.

It is also possible to execute the segmentation of the target phase from CT scan data independent of Avizo® software by parsing and processing the image slices

independently using software tools developed inhouse. We have designed an alternative way to pre-process the microtomographic data by (1) reading the greyscale values directly from image files; (2) cropping the proper parallelepiped by specifying the boundary position in 3-directions; and then (3) exporting the similar label-data by comparing the value of each voxel with a user selected threshold value. This pre-processing stage has the advantage of being able to be run on high performance computers. Thus it can deal with larger datasets than commercial products and is not constrained by license requirements.

### 40.2.3 Examples

In the following we present some examples of microtomography.

Figure 40.1 shows pictures of a strongly deformed rock (mylonite) sample. Figure 40.1a is the hand specimen and the position of one sub-sample is marked (a total of 54 sub-samples have been scanned). Figure 40.1b is one slice of the synchrotron CT scan from the marked sub-sample. Each slice has $2048 \times 2048$ pixels at a resolution of $1.3\,\mu m$. The shape is a square in this slice of the sub-sample and the corners are cut off by a circle. Non-material pixels are those that extend beyond the overlap of the square and the circle, these should not be included in the exported label-data that will subsequently be used for quantitative analysis. Using the naked eye, we can recognise tiny pores, light-grey grains, a medium-grey material, and a white mineral. These can be identified as the target phase and analysed quantitatively. Figure 40.1c is the 3-D binary model of pores in the volume consisting of $1300 \times 1300 \times 1550$ voxels. The visualisation of the pore-structure illustrates how the pores are distributed in space, they are irregularly distributed and small pores tend to be are grouped together.

Figure 40.2 is a gypsum sample that is heated from room temperature to $235\,°C$. The description of the experiment is given in Fusseis et al. (2010b). As it is a laboratory made gypsum sample, it does not contain inner structure such as grains and pores but it does have some impurities that appear as white specks in the images. Being heated, the dehydration of gypsum leads to the creation of cracks, and the cracks propagate from the perimeter to the center of the sample. Figure 40.3 shows the cracks of the final step in a volume of $400 \times 800 \times 300$ voxels. Separated cracks are shown in different colours. We can see that the structure of the cracks is very complex, with thin, curved, and discontinuous features. Through quantitative analysis we can parameterise the size, shape, and orientation of each crack.

Figure 40.4 is an example of a tree branch. A portion of the CT scan is shown in Fig. 40.4a. The CT-scan slice is $2048 \times 2048$ pixels with a resolution of $2.88\,\mu m$. As can be seen in Fig. 40.4b there are a large number of pores, these are conduits of the tree branch. Figure 40.4c is a $288\,\mu m^3$ volume ($100^3$ voxels) and shows some conduits that are connected by sideways-connected cells (pit cells). Connected conduits are denoted by the same colour.

**(a)**



**(b)**

**(c)**
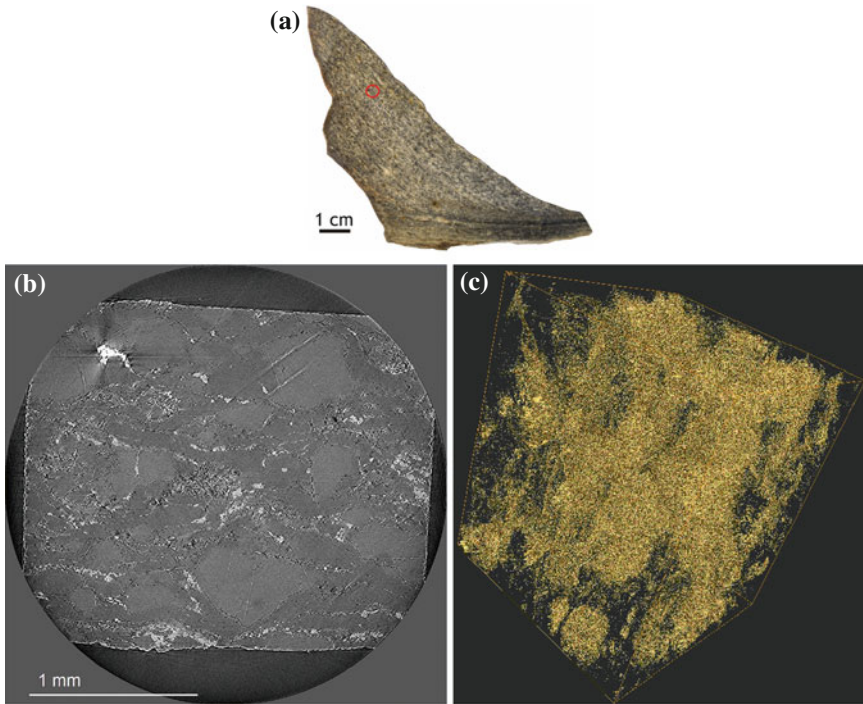
**Fig. 40.1** Pictures of a strongly deformed rock (mylonite) sample, hand specimen and the position of a sub-sample (**a**), a slice of the sub-sample (**b**), and the 3-D pore-structure of $1300 \times 1300 \times 1550$ volume presented as an iso-surface using Avizo® (**c**)
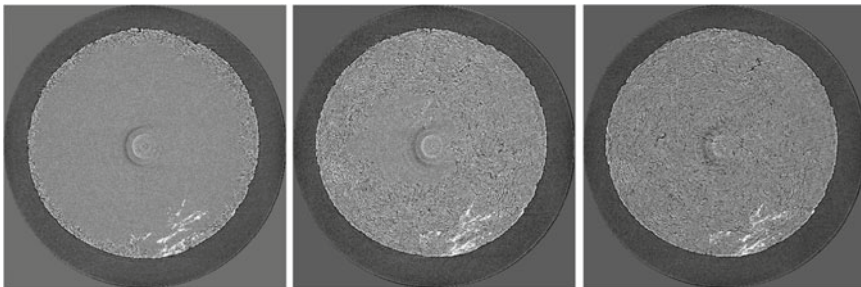


**Fig. 40.2** A gypsum sample being heated, the same slice is shown at three time steps. Resolution is the same as in Fig. 40.1

## 40.3 Methodology

We use percolation theory for the analysis of microtomography. The mathematical definition of percolation refers to the nature of the connectivity in lattice models. Percolation theory was developed to analyse the connectivity of sites/bonds of
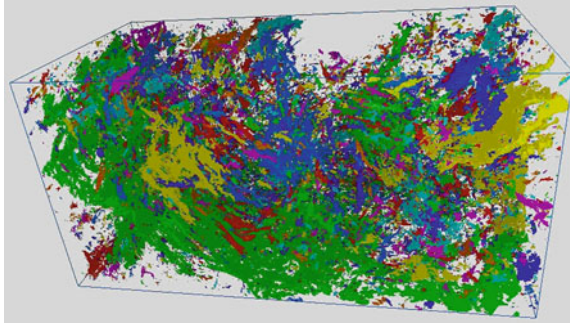
**Fig. 40.3** 3-D visualisation of dehydrated cracks of the gypsum sample in the final step of a volume of $400 \times 800 \times 300$ voxels. Separated cracks are shown in different colours
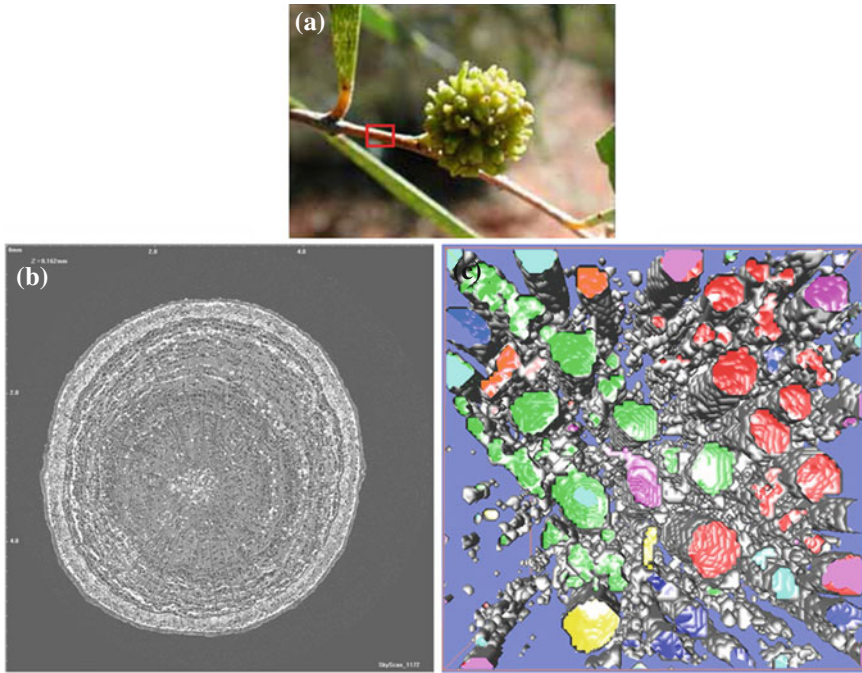


**Fig. 40.4** A tree branch sample, tree specimen (**a**), a slice of CT-scan image in inverted colour (**b**), and a small volume showing conduits (**c**)

different lattice models, critical phenomena, and their related quantities (Stauffer and Aharony 1994). The 3-D binary model after segmentation can be considered as a simple cubic lattice model in which every cubic cell (or site) is equivalent to a voxel in a volumetric dataset.

### 40.3.1 Basic Output Parameters

The principal parameter of our quantitative analysis is the volume fraction of the target phase, which is, for example, porosity $\varphi$ in porous media. It is defined as the ratio of the volume of target phase and the total (or bulk) volume. Both volumes can be determined from the label-data of the 3-D model. The porosity of a rock is, for instance, an important quantity for the evaluation of the potential volume of water or hydrocarbons it may contain.

The specific surface area (SSA) $s_p$ is defined as,

$$s_p = S_p/V, \tag{40.1}$$

where $S_p$ represents the surface area of target phase and $V$ is the total volume. Surface area is equivalent to the iso-surface of volumetric data processing that separates the target phase from the matrix. As a voxel in cubic lattice model is recognized as a hexahedron, the surface area can be directly derived from the label-data. Note here $s_p$ has the unit of $a^{-1}$ (or $1/a$), where $a$ is the lattice constant, i.e. the resolution of the images. The SSA is a derived quantity that can be used to determine the type and properties of a material. It has a particular importance in case of adsorption, heterogeneous catalysis, and reactions on surfaces.

For a granular material the distribution of grain size is called the particle size distribution (PSD). The PSD of a material can be important in understanding its physical and chemical properties. It affects the strength and load-bearing properties of rocks and soils, and the reactivity of solids participating in chemical reactions. PSD can also be referred to as pore size distribution when the pores are the target phase. They are obtained on the basis of stereology by choosing arbitrary planes parallel to $x$, $y$ and $z$ axis and counting the particles (or pores). For a plane parallel to $x$ axis, for example, we count the target phase in different sizes in the $x$ direction on the plane. After counting other planes parallel to $x$ axis, the summations of different sizes of the target phase and their percentages in the $x$ direction are calculated. A similar procedure is performed in the $y$ and $z$ directions.

### 40.3.2 Clusters and Orientations

In the simple cubic lattice model, the nearest-neighbours are voxels with one common plane. A cluster is a group of nearest-neighbours of the same material that are connected to each other. Labelling clusters is a process of giving all cells within the same cluster the same label. The Hoshen-Kopelman algorithm (Hoshen and Kopelman 1976) is used to reduce the computing time in this procedure. After the cluster-labelling, each voxel is assigned a value representing the cluster number it belongs to. The largest cluster (with maximum voxel number) is cluster 1, the second largest one is cluster 2, and so on. The dataset is known as cluster-labelled

data to distinguish it from the binary material-labelled data resulting from the segmentation.

Percolation of a model can be determined after cluster labelling. Since all clusters in the model now have their specific label, it is possible to know which labels appear in the outer boundaries. We define a direction as percolating when one boundary of the parallelepiped in this direction has the same cluster label with its opposite boundary. This means the two end boundaries of the direction are connected by the same cluster and the model is permeable.

The shape and orientation of a cluster can be described by a tensor. The tensor is obtained by using the star volume distribution method following Ketcham (2005). For the cluster that consists of a set of $n$ sites (voxels), each site $i$ is considered as a vector $\mathbf{a}_i = (a_{xi}, a_{yi}, a_{zi})^{\mathrm{T}}$ relative to the center of the cluster. The shape and orientation of the cluster can be defined by the following matrix

$$\mathbf{T} = \sum_{i=1}^{n} \mathbf{a}_i \mathbf{a}_i^{\mathrm{T}}. \tag{40.2}$$

The matrix $\mathbf{T}$ has 3 eigenvalues $\tau_1 < \tau_2 < \tau_3$ and corresponding eigenvectors $\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3$. The direction cosines of the eigenvectors describe the orientation of the cluster. The matrix can be visually represented by an ellipsoidal glyph. Isotropy index $I = \tau_1/\tau_3$ and elongation index $E = 1 - \tau_2/\tau_3$ describe the shape of the cluster in a simplified way. For porous media, if we assume that permeability anisotropy is relevant to the character of the pore structure, then the matrix $\mathbf{T}$ denotes the anisotropy of permeability.

### 40.3.3 Stochastic Analysis

Stochastic analysis is employed to detect the effect of scale and is computed by using the extended local porosity theory (Liu et al. 2009; Hilfer 1992, 2002). To implement the local porosity theory a cubic sub-volume with side-length $L$ acts as a moving window (Williams and Baxter 2006) scanning through the whole model. Each placement of the moving window is denoted as K($\mathbf{x}$, $L$), where $\mathbf{x}$ is the central position of the sub-volume. The window offset can be equal to the lattice constant or an integer multiple of it. For every placement of the sub-volume, porosity (volume fraction), percolation, and anisotropy of the percolating cluster are calculated. Statistical results can then be drawn from the calculations of all placements of the sub-volume.

The statistical results are given by three probabilities.

(1) The local porosity distribution $\mu(\varphi, L)$ is defined as

$$\mu(\varphi, L) = \frac{1}{m} \sum_{\mathbf{x}} \delta(\varphi - \varphi(\mathbf{x}, L)), \tag{40.3}$$

where $\delta$ denotes the Dirac $\delta$-function, $\varphi(\boldsymbol{x}, L)$ is the local porosity in the sub-volume $K(\boldsymbol{x}, L)$, $m$ is the total number of placements of the moving window. $\mu(\varphi, L)$ is the proportion of the number of placements with the porosity $\varphi$ and of the sub-volume-size $L$.

(2) The local percolation probability in the $\alpha$-direction is defined through

$$\lambda_\alpha(\varphi, L) = \frac{\sum_{\boldsymbol{x}} \Lambda_\alpha(\boldsymbol{x}, L)\delta_{\varphi,\varphi(\boldsymbol{x},L)}}{\sum_{\boldsymbol{x}} \delta_{\varphi,\varphi(\boldsymbol{x},L)}}, \tag{40.4}$$

where

$$\delta_{\varphi,\varphi(\boldsymbol{x},L)} = \begin{cases} 1 & \text{if } \varphi = \varphi(\boldsymbol{x}, L) \\ 0 & \text{otherwise} \end{cases},$$

$$\text{and} \quad \Lambda_\alpha = \begin{cases} 1 & \text{if percolating in } \alpha\text{-direction} \\ 0 & \text{otherwise} \end{cases},$$

$\alpha = x, y$, or $z$ denotes the direction along every axis, and $\alpha = 3$ represents the percolation along all three directions. It is the proportion of the percolating placements of the sub-volume-size $L$ and porosity $\varphi$.

(3) The local anisotropy distributions are defined as

$$\chi(I, L) = \frac{1}{m'} \sum_{\boldsymbol{x}} \delta(I - I(\boldsymbol{x}, L)), \tag{40.5}$$

where $m'$ is the total number of percolating sub-volumes, $I$ represents two indices of anisotropy, the isotropy index $I_1 = I = \frac{\tau_1(\boldsymbol{x},L)}{\tau_3(\boldsymbol{x},L)}$ and the elongation index $I_2 = E = 1 - \frac{\tau_2(\boldsymbol{x},L)}{\tau_3(\boldsymbol{x},L)}$. $\chi(I, L)$ gives the proportion of the placements with the index value $I_i$ and the sub-volume-size $L$. Here $\tau_i(\boldsymbol{x}, L)$ are three normalised eigenvalues of the orientation matrix of the percolating cluster in sub-volume $K(\boldsymbol{x}, L)$ that satisfy $\sum_{i=1}^{3} \tau_i(\boldsymbol{x}, L) = 1$. When more than one cluster is percolating in a sub-volume, only the largest percolating cluster is considered for the statistical analysis.

The stochastic analysis has two important goals. Firstly, the three probabilities are scale-dependent, that is, for different scales or sub-volume-sizes $L$, there are different distribution functions. When all these probabilities converge at a given scale the size of RVE is determined. Secondly, combining the probabilities of porosity, percolation and anisotropy of microtomography with the statistical moment method allows the generation of derivative models that retain the original material structure. Thus, we can create digital samples that represent the characteristics of the microstructure. Digital samples provide another way of upscaling, but it will not be discussed in this paper.

## *40.3.4 Scaling Parameters and Upscaling Scheme*

In this section the definition and computation of scaling parameters are given, such as percolation threshold, critical exponent of correlation length, fractal dimension. The strategy of upscaling is also explained briefly. A more detailed description and applications can be found in Liu and Regenauer-Lieb (2010).

### 40.3.4.1  Percolation Threshold

In percolation theory (Stauffer and Aharony 1994), concentration $p$ is the probability of a site belonging to the target phase; it is equivalent to volume fraction for the whole model. The (critical) percolation threshold $p_c$ is the minimum concentration at which percolation would occur in the model. To determine the percolation threshold it is essential to have models with similar structures but different volume fractions. From static microtomographic data the percolation threshold is difficult to find for the specific structure.

We have developed a morphological technique to determine the percolation threshold of a microstructure by deflating (shrinking) and inflating (expanding) the target phase. As in 3-D binary models, the target phase is labelled as 0 and the matrix is labelled as 1. A deflation/inflation operation involves modifying the labels of voxels in the model, i.e., deflation changes a label from 0 to 1 for any target-phase-voxel that has one or more nearest-neighbours belonging to the matrix, inflation changes label from 1 to 0 for any matrix-voxel when it has one or more nearest-neighbours belonging to the target phase. Only nearest-neighbours are considered when modifying labels. This is consistent with the definition of clusters in the percolation analysis. The operations of inflation and deflation are manipulated iteratively from the original model to create inflated and deflated models, respectively. In this way a series of derivative models are created. These models have different volume fractions but retain structures similar to those in the original sample. By analysing the percolation of these models the critical model is identified as the percolating model closest to the non-percolating model. The volume fraction of the critical model is recognized as percolation threshold (Liu and Regenauer-Lieb 2010).

### 40.3.4.2  Critical Exponent of Correlation Length

There are several different critical exponents in percolation theory that describe some quantities that are divergent when concentration $p$ is close to the percolation threshold $p_c$. The significance of critical exponents is that critical exponents are related to scaling laws and scaling laws enable up-scaling/down-scaling of the characteristics identified at a particular scale. Only two critical exponents are independent and in most cases the critical exponent of correlation length is the easiest to calculate.

The correlation length $\xi$ defines the average distance between any two sites belonging to the same cluster (Stauffer and Aharony 1994). It is defined as

$$\xi^2 = \frac{2\sum_s R_s^2 s^2 n_s}{\sum_s s^2 n_s},$$ (40.6)

where $s$ is the number of sites of a cluster, $n_s$ is the number of such $s$-site clusters per lattice site, $R_s$ defined as

$$R_s^2 = \sum_{i=1}^{s} \left(|x_i - x_0|^2 /s\right)$$ (40.7)

is the radius of a complex cluster of $s$-site; where $x_0 = \sum_{i=1}^{s} (x_i/s)$ is the position of the centre of mass of the cluster, and $x_i$ is the position of the $i$th site in the cluster. The correlation length $\xi$ diverges when volume fraction $p$ approaches the percolation threshold $p_c$, as

$$\xi \propto |p - p_c|^{-\nu}.$$ (40.8)

Here $\nu$ is the critical exponent of correlation length (or correlation length exponent). It can be extracted by a finite-size scaling scheme from the local percolation probabilities $\lambda(p, L)$ of different sub-volume-size $L$ and different volume fraction $p$ (Liu et al. 2009; Hilfer 1992, 2002).

### 40.3.4.3 Fractal Dimension

The fractal dimension $D$ is derived from a power law describing a characteristic versus a statistical variable. In percolation theory the fractal dimension is included in scaling laws with critical exponents. The fractal dimension can be found from the relative size and the number of clusters defined as

$$D = \frac{\log_{10} N \left(l \geq \dfrac{R}{R_{\max}}\right)}{\log_{10} l \left(= \dfrac{R}{R_{\max}}\right)},$$ (40.9)

where the relative size of cluster $l$ is the radius of Eq. (40.7) normalised by the largest cluster radius, the number of cluster $N$ includes clusters that are equal to or larger than the normalised relative radius.

### 40.3.4.4 Strategy of Upscaling

With all above listed parameters defined and extracted from the micro-CT we can move on to the upscaling scheme.

The stochastic analysis provides probabilities of porosity, percolation and anisotropy at different scales (limited to the 3-D microstructure model), and can be used to determine the size of the RVE. Different numerical simulation methods, such as finite element, finite difference and lattice-Boltzmann method, can be used to compute material properties from microstructures. Simulations should be based on the RVE, since only the RVE can represent the general characteristics. The upscaling workflow involves (1) cluster analysis using Eq. (40.6) to derive the fractal dimension $D$; (2) finite-size scaling scheme using the probability of percolation to derive the critical exponent of correlation length $\nu$; (3) inflation/deflation of original structure to determine percolation threshold from a series of derivative models. The critical model with percolation threshold provides independent way to extract fractal dimension $D$ and the crossover length $\xi$. The crossover length is a special correlation length separating the critical behaviour and non-critical behaviour. When we obtain the critical exponent of correlation length $\nu$ and fractal dimension $D$ from individual samples, the final derived scaling laws are exact for the specific structures.

## 40.4 High Performance Computing

Computationally, the percolation analysis described by Eqs. (40.3)–(40.10) is implemented in Fortran90. Test and validation of the implementation has been reported in Liu et al. (2009), Liu and Regenauer-Lieb (2010). We have developed two main codes. One is "PROP" for stochastic analysis. In PROP only the largest percolating cluster in a sub-volume is analysed concerning the anisotropy in statistics as described in Sect. 40.3.3. The other code "CTSTA" accomplishes the analysis and output of all clusters in an entire model. Statistics of individual clusters in a model provides detailed information of the inner structure and can also be used to calculate the fractal dimension of the microstructure. Both codes are written for high performance computers.

### 40.4.1 Parallelisation of Stochastic Analysis

The stochastic analysis of the moving window method in PROP is the most time consuming computation. For a single volume, the analysis of porosity, percolation and anisotropy can be calculated on a single CPU in a short amount of time, e.g., 0.1–0.2 s for a volume of 1 million voxels. However, the total number of sub-volumes $m$ in a stochastic analysis is given by

$$m = \prod_{i=1}^{3} \left( \text{INT} \left( \frac{M_i - L}{n} \right) + 1 \right), \tag{40.10}$$

where $M_i$ is the side-length of the model in units of the lattice constant $a$, i.e. the resolution of the images, $n$ represents the moving step in units of the lattice constant. When $n = L$, there is a non-overlapping placement of sub-volumes that can lead to fluctuation in the results. For $n < L$, there is an overlapping placement and there is a higher weight given to the central region of the model. For a large data set and a small $n$ the computing time on a single processor becomes increasingly problematic. To alleviate this problem, it is necessary to parallelise the calculation.

### 40.4.1.1 OpenMP Version (PROP_omp)

The probabilities defined in Eqs. (40.3)–(40.5) are dependent only on the summation of data within a moving window. By using the OpenMP programming techniques it was possible to quickly and easily augment the DO loops that move this window to move a series of windows, one for each available CPU. This parallelisation distributes the most time consuming part of the calculation cross multiple CPUs. As shown in Liu et al. (2009), the speedup is quite good when fewer than 32 CPUs are used.

The usability of PROP_omp is limited to shared memory processing (SMP) systems. Subject to the number of cores and physical memory available, PROP_omp can perform reasonably well only for medium size of sub-volume. Suppose each voxel in the sub-volume is represented by a 4 byte integer, the maximum theoretical sub-volume size is around $812^3$ voxels for 2GB of RAM. In practice it is less than this as storage is also required for additional data structures, e.g., sub-volume-size can be from $500^3$ to $640^3$ depending on the volume fraction of the target phase. In addition, large-scale SMP systems consisting of 1000+ cores are becoming available for high performance computing. However due to limitations of the OpenMP architecture it is very difficult for PROP_omp to scale to hundreds of cores.

### 40.4.1.2 MPI Version (PROP_mpi_1)

To address these issues we then have parallelized PROP with message passing interface (MPI) in order to perform the stochastic analysis on larger sub-volumes (in the order of $1000^3$ voxels) on a distributed memory machine and ensure the analysis runs in a massively parallel manner. The parallelisation consists of three major tasks: (1) flexibly assign or distribute the sub-volumes described by Eq. (40.10) onto different computing cores, (2) efficiently deal with input and output and (3) efficiently collect the statistics of the sub-volumes.

The second and third tasks are relatively straightforward, i.e, the parallel I/O is achieved through linking the I/O channels to MPI rank ID. The collection of the statistics over the sub-volumes is achieved with the MPI_Reduce function. The first task requires a careful design of a mapping topology between the order of sub-volumes and the number of computing cores. The mapping topology is chosen such that one can run PROP_mpi_1 with the number of cores being a factor of $N_3$ or $N_3 \times N_2$, where $N_3 = 1 + (M_3 - L)/n$ and $N_2 = 1 + (M_2 - L)/n$, assuming there

are a sufficient number of cores on a massively parallel system. One can also flexibly run PROP_mpi_1 with any number of MPI processes on a small Linux clusters with load imbalance minimised if sufficient number of cores are not available.

The results of PROP_mpi_1 have been compared with those generated by PROP_omp for the same set of data ($469 \times 657 \times 626$). Identical results are obtained (up to 5th digit), indicating the cumulative error associated with the use of MPI_Reduce is negligible. It is worth mentioning that the use of the MPI_Reduce function is a global operation and requires all MPI processes to send the data to the master. The scalability of PROP_mpi_1 with a large number of cores can be compromised if the statistical variables are not packed optimally or too many calls of MPI_reduce are used. Scalability tests of PROP have been performed on up to 540 cores, showing a linear speedup.

## 40.4.2 Parallelisation of Stochastic Analysis and Hoshen-Kopelman Algorithm (PROP_mpi_2)

The two versions of PROP code discussed above are based on the idea of analysing one sub-volume per CPU core. The calculation thus requires large physical memory when the size of sub-volume increases. Processing even larger volumes requires overcoming the physical memory constraint, we must distribute our analysis of a single sub-volume across multiple CPU cores. While this is more efficient use of RAM it is expected to be less efficient from a pure speed point of view as data dependencies will be introduced necessitating that the CPU cores coordinate their analysis.

Decomposition of sub-volume data proceeds with a technique commonly used when dealing with 3D regular grids. That is, if we have two CPUs we split the data in half along the $x$ axis, 4 CPUs splits both the $x$ and $y$ axis, and 8 CPUs splits the $x$ $y$ and $z$ axis. A similar process can be applied if more CPUs are available. We also allocate a "halo" of voxels, so that each CPU can receive information about voxels on the other side of the split.

The analysis proceeds by applying the Hoshen-Kopelman algorithm to each fragment of the sub-volume individually. Each CPU begins numbering clusters in its fragment of the sub-volume at "1". At this point it is not possible to identify if a cluster on one CPU is the same as a cluster on a neighbouring CPU. Nor is it possible to identify if two clusters on one CPU are actually joined through a cluster on an adjacent CPU.

Once the Hoshen-Kopelman algorithm has completed on each CPU, the CPUs communicate their number of clusters found and clusters are re-labelled to give them unique cluster labels. That is, if CPU 1 found 10 clusters then CPU 2 will add 10 to each of its cluster labels. If CPU 1 found 10 clusters and CPU 2 found 8 clusters then CPU 3 will add 18 to each of its cluster labels.

Each CPU will now send to its neighbours (defined by the $x$ $y$ and $z$ axes) the cluster labels associated with voxels on its boundary. In this way each CPU can calculate which of its clusters are in direct contact with clusters on a neighbouring CPU. Contact between two clusters is defined by what amounts to an entry in something similar to a Compressed Sparse Row (CSR) representation of the adjacency matrix of the graph. That is, an array is allocated of $m$ rows by (number of clusters) columns, where $m$ is the maximum of number of clusters that are likely to be touching. Each CPU fills in the columns associated with the clusters it stores locally. The information in the adjacency matrix is shared between all CPUs by a call to MPI_Allreduce. We note that the CSR representation is inefficient in RAM when the correlation length is much less than the side length of the sub-volume fragment. If the correlation length is short, most clusters will not touch a side of their sub-volume fragment and will be guaranteed not to have neighbours. In this case, the CSR representation will allocate space for column with no rows.

Using the computational infrastructure presently available to us we have not run a calculation large enough to find the upper limit of the code. We predict that as the sub-volume includes more voxels the CSR representation will the limiting factor in the calculation size.

Finally we can use a standard Depth First Traversal (DFT) of the graph to create a mapping between old cluster labels and new cluster labels. A depth first traversal means that each cluster has a list of neighbours. We pick a cluster and relabel its first neighbour to point to this cluster. We then relabel the first neighbour's first neighbour. If this cluster has no subsequent neighbours we can move up to the first neighbour's second neighbour. Eventually we will move onto the subsequent neighbours of our original node. The same depth first traversal is performed on all CPUs used in the calculation thus avoiding the cost of communicating the results of the DFT between all CPUs. Further work on the parallel Hoshen-Kopelman implementation would involve replacing the CSR representation with a more RAM efficient representation and performing the graph traversal in parallel.

### 40.4.3 Calculation of Individual Clusters

To calculate the position, shape and orientation of every cluster might be troublesome when there are a great number of clusters in a volume. The problem arises from the large number of clusters and the large differences in cluster sizes. As defined in Eq. (40.2), to calculate $T$ we firstly need to define arrays to load all the vectors of $\boldsymbol{a}_i = (a_{xi}, a_{yi}, a_{zi})^{\mathrm{T}}$ of clusters.

There are two options, either one array loads all clusters or there is one array per cluster. For the former option no matter how many dimensions of the array we define the size of the dimension must be sufficient to load the largest cluster. That means that when the smallest array (1 voxel) is being calculated it is using the same size of array and the same memory as the largest one. For the later option, as it is impossible to define millions of arrays in the code loading an uncertain numbers of clusters, we

can only define a dynamic array to load $\boldsymbol{a}_i = (a_{xi}, a_{yi}, a_{zi})^{\mathrm{T}}$ for only one cluster at a time. That means we need to read the 3-D cluster-labelled data each time for every cluster, which is unacceptably inefficient.

We cope with this problem by grouping the clusters, which is similar to data decomposition. A certain amount of memory is allocated to load the vectors $\boldsymbol{a}_i$. Since the dimension-size of the array always fits the largest cluster in the group, when large clusters are being calculated the group has fewer clusters; then when small clusters are being processed, the group can include a great number of clusters. This idea balances memory usage and computing time and results in a high performance solution.

## 40.5  Applications

The capabilities of our method have been verified and presented as specific case studies in Liu et al. (2009), Liu and Regenauer-Lieb (2010). Here we give some more unpublished examples and expand on the capabilities. Some post-processing techniques and visualisation are also described in the examples.

The following presents the results of the sample introduced in Fig. 40.1c. The porosity is 1.23 %; the model is not percolating; the SSA is 0.0327; the pore size distributions in three directions are shown in Fig. 40.5. It shows that more than 90 % pores are smaller than 9 μm (7 voxels) in all $x$, $y$ and $z$ directions. The largest pore is 70 μm (54 voxels) in $z$ direction and it is the only one (out of the range of $x$ axis of Fig. 40.5).

There are in total 1,895,341 (∼1.9 million) clusters in the volume. Nearly 1 million clusters are 1 or 2 voxel-clusters. The distribution of cluster size and the percentage of each cluster size over all pores are shown in Fig. 40.6. The number of clusters linearly decreases with increasing cluster size on a double logarithmic plot. Large clusters are unique and also exhibit a linear distribution along the $x$ axis of Fig. 40.6. Interestingly, the large number of small clusters and the few large clusters occupy
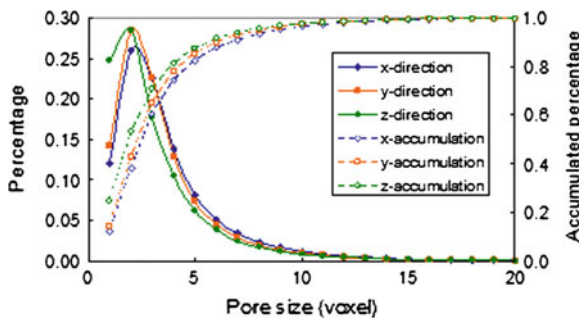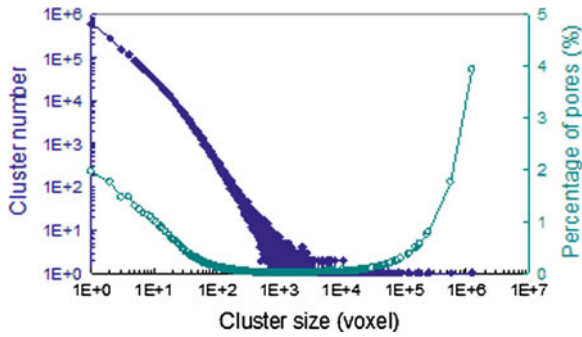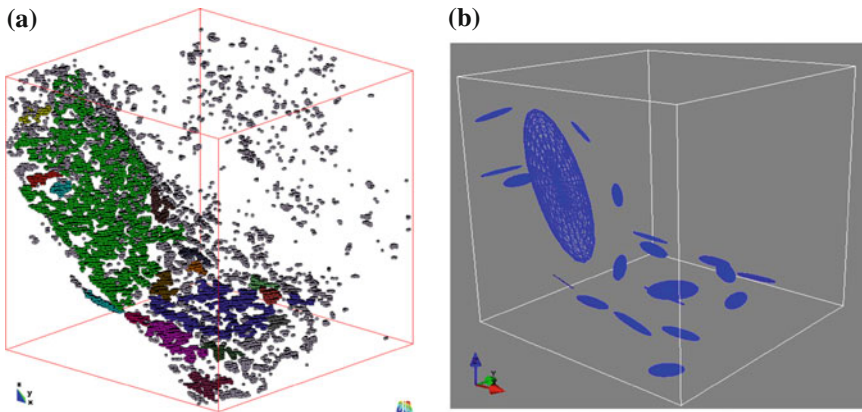


**Fig. 40.5** Pore size distribution of the strongly deformed rock sample

**Fig. 40.6** Histogram of cluster size of the sample in Fig. 40.1, and the volume percentage of different cluster-sizes

**(a)**                                              **(b)**



**Fig. 40.7** Display of clusters in a 100 cube volume of the sample in Fig. 40.1

a high percentage of the volume of all pores; in contrast, middle-size clusters (with 100–10000 voxels) occupy a low proportion of the pores in the model. The largest cluster occupies less than 4 % space of the porosity.

Visualisation of clusters illustrates the characteristics of each cluster of the target phase in a model, which provides more information than visualisation of the target phase itself (as shown in Fig. 40.1c). Figure 40.7a shows the structures of clusters in a small volume of the same sample using GID®, in which each voxel is presented by a hexahedron. Different clusters are differentiated by using different colours but all tiny clusters are presented in one colour. Figure 40.7b illustrates the corresponding orientation tensors of the 20 largest clusters presented by ellipsoidal glyphs using Mayavi 2. Glyphs are located in the center of the cluster, and the radii are proportional to the sizes of the clusters. In this small volume pores are apparently mostly distributed along a crack but the connectivity of pores is poor and the volume is not percolating.
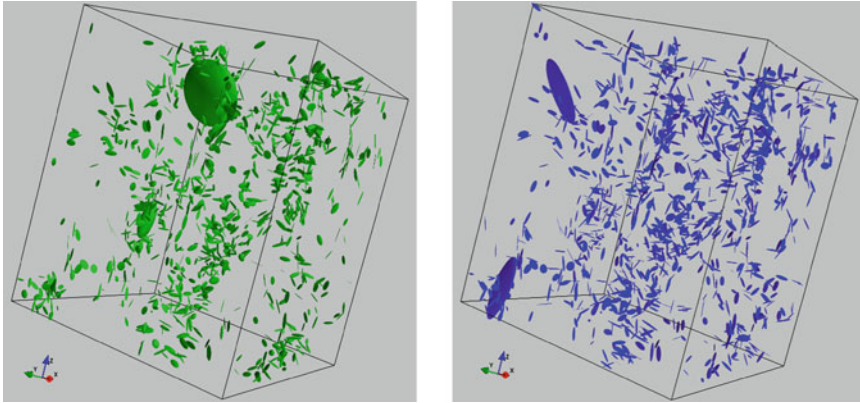
**Fig. 40.8** Selected clusters of the sample in Fig. 40.1, the 1000 largest clusters (*left*) and large strongly anisotropic clusters that index ≤0.1 and cluster size ≥1000 voxels (*right*)

Although our code can deal with very large datasets the visualisation of large datasets can be limited by software. Some small post-processing codes were written to help to maximize the ability to meaningfully visualize the data. For instance, the orientation of very small clusters, especially 1- or 2-voxel-clusters, is generally not important, thus we can choose to only include the large clusters. We also can select which clusters to display according to their positions, size, isotropy index, elongation index, or combinations of these quantities. Figure 40.8 shows the selected clusters of the sample in Fig. 40.1, the 1000 largest clusters (left) and large strongly anisotropic clusters with isotropy index ≤0.1 and cluster size ≥1000 voxels (right). Both of these two groups of clusters illustrate the general character of the distribution of pores of the sample.

The histograms of the isotropy index and elongation index of the same sample are shown in Fig. 40.9. We can see that the isotropy index of most clusters is less than 0.2 and are more concentrated between 0.03–0.13. The elongation index of most clusters is larger than 0.5 and peaks at around 0.8. Both these two indices suggest that most pores in the sample are strongly anisotropic. Note that the statistics in Fig. 40.9 do not include clusters ≤20 voxels. The reason is that the orientation tensor is calculated by summing the dyadic product of the site vector, see Eq. (40.2). When the cluster is small each site vector has a major influence on the orientation tensor. The sensitivity of the site vector to orientation tensor places a lower limit on the cluster size for the meaningful statistic analysis. For example, there are high counts of 1, 2 and 3 voxel-clusters that generate an extremely high fraction of isotropy index of 1, 0.5, and 0.33, which should be ignored in the statistics.

For the applications of stochastic analysis the following examples and applications have been published. (1) Liu et al. (2009) gave a synthetic sandstone sample as the benchmark. The probabilities of porosity, percolation and anisotropy all asymptotically converge when the sub-volume-size approaches to 400 voxels (1 mm),
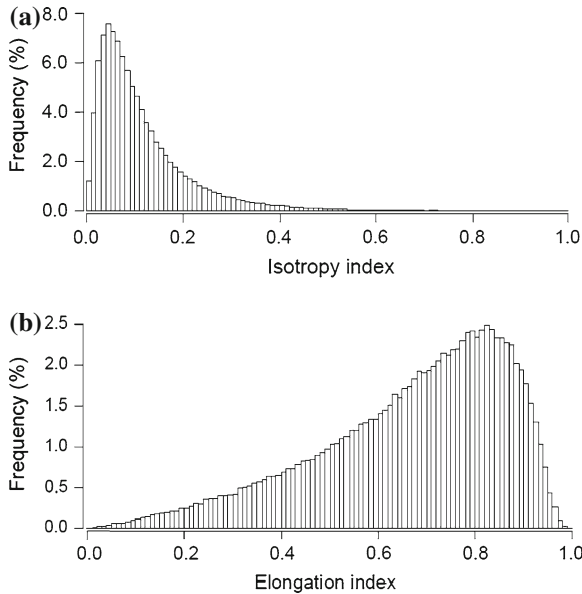
**Fig. 40.9** Histogram of isotropy index (**a**) and elongation index (**b**) of the sample of Fig. 40.1

thus the size of RVE of this sample was determined. The RVE was later used to calculate the permeability and the result was comparable with experimental laboratory result (Liu and Regenauer-Lieb 2010). (2) Liu et al. (2010) gave two sub-samples from the same hand specimen of Fig. 40.1. These two sub-samples showed different characteristics from the synthetic sandstone sample and they are different from each other. As the probabilities of anisotropy showed fluctuation the size of RVE cannot be determined. (3) Liu and Regenauer-Lieb (2010) also gave some results of a tree branch sample. This application showed that percolation theory can be used in analysis of this type of strongly anisotropic model, but the extraction of correlation length and critical exponent of correlation length are unattainable.

The scaling parameters and upscaling scheme is our latest progress and mostly presented in Liu and Regenauer-Lieb (2010). It uses the results of PORP and CTSTA computations and combines some new approaches such as inflation/deflation of images, calculation of fractal dimension, and supplementary analysis. It is based on the visualisation, characterisation and high performance computing of microtomography and makes it possible to extend the scale from micro-scale to macro-scale. The relevant discussion is beyond the scope of the present contribution. For a more detailed description please refer to the aforementioned paper.

## 40.6 Conclusion and Discussion

Our method and computer algorithms have achieved (1) computation of major parameters, including volume fraction, specific surface area, particle (or pore) size distribution; (2) stochastic analysis and determination of RVE, probabilities of porosity, percolation, and anisotropy for different scales are computed, when these probabilities are convergent at a scale the size of RVE is determined; (3) extraction of critical exponent of correlation length, fractal dimension, and percolation threshold, thus parameters of scaling laws are determined making it possible to scale up properties from micro-scale to macro-scale. With these three capabilities, our method is unique, more comprehensive, and more powerful than other commercial/published codes for the analysis of microtomography.

We are now applying this methodology to routinely derive elastic percolation, electrical percolation, thermal conductivity and many other properties from CT-scans. Our analyses showed that for samples with strongly anisotropic structure the critical exponent of correlation length is not attainable (Liu and Regenauer-Lieb 2010). In this situation other methods for achieving the upscaling must be used. Our long-term goal is to establish an integrated methodological system with applications to multi-scale system dynamics. The progress in this paper is a milestone that not only provides a tool for detailed analysis of microtomography, but also links the micro-scale to macro-scale.

Considering the size of microtomographic data and the degree of computation required, the main code has been progressively parallelised. The first stage of the parallelisation is based upon OpenMP, which performs very well for most situations. Limited by the OpenMP architecture the performance does not continue to scale as more CPUs are used. The development of the second stage of parallelism was based on MPI. This version removes the limitation of 32 CPUs and the performance improvement is linear up to at least hundreds of CPUs. However, this version does not consider the decomposition of data, thus the size of datasets is still a barrier. To improve this aspect of the code a new version, also based upon MPI, has been developed which includes data decomposition. By distributing the whole dataset and sub-volumes into CPUs the size of datasets is no longer a barrier. This third version has additionally achieved a significant breakthrough by parallelising the Hoshen-Kopelman algorithm.

Although all the progress and capabilities have been very successful there is still significant scope for further improvements. We believe the performance of the latest MPI version can be improved further. It is also acknowledged that GPU computing has been maturing over the last few years and is showing great advantages in computing speed and energy efficiency. We expect the algorithms presented here to be suited to GPU computing and thus significant performance gains should be achievable.

# References

Fusseis F, Schrank C, Liu J, Karrech A, Llana-Fúnez S, Xiao X, and Regenauer-Lieb K (2012) Pore formation during dehydration of polycrystalline gypsum observed and quantified in a time-series synchrotron x-ray tomography experiment. Solid Earth 3: 71-86

Hilfer R (1992) Local porosity theory for flow in porous media. Phys Rev B 45:7115–7121

Hilfer R (2002) Review on scale dependent characterization of the microstructure of porous media. Transp Porous Med 46:373–390

Hoshen J, Kopelman R (1976) Percolation and cluster distribution 1. Cluster multiple labelling technique and critical concentration algorithm. Phys Rev B 14:3438–3445

Ketcham RA (2005) Three-dimensional grain fabric measurements using high-resolution X-ray computed tomography. J Struct Geol 27:1217–1228

Lindquist WB (2005) 3DMA-Rock. http://www.ams.sunysb.edu/~lindquis/3dma/3d_imaging.html. Accessed 13 Dec 2010

Liu J, Regenauer-Lieb K (2010) Application of percolation theory to microtomography of structured media: percolation threshold, critical exponents and upscaling. Phys Rev E (in press)

Liu J, Regenauer-Lieb K, Fusseis F (2010) Stochastic analysis of percolation and anisotropic permeability from microtomographic images and an application to mylonites. In: Satake K (ed) Advances in geosciences, vol 20. World Scientific Publishing Co., Singapore, pp 229–245

Liu J, Regenauer-Lieb K, Hines C, Liu K, Gaede O, Squelch A (2009) Improved estimates of percolation and anisotropic permeability from 3-D X-ray microtomographic model using stochastic analyses and visualization. Geochem Geophys Geosyst 10:Q05010. doi:10.1029/2008GC002358

Nakashima Y, Kamiya S (2007) Mathematica programs for the analysis of three-dimensional pore connectivity and anisotropic tortuosity of porous rocks using X-ray computed tomography image data. J Nucl Sci Technol 44:1233–1247

Stauffer D, Aharony A (1994) Introduction to percolation theory, 2nd edn. Taylor & Francis Ltd., London

Williams TO, Baxter SC (2006) A framework for stochastic mechanics. Probab Eng Mech 21:247–255