공학석사학위논문

Morphological Feature-Based Microstructure Characterization and Reconstruction of Mechanoluminescent Composites through Particle Shape Library

형상 라이브러리를 이용한 응력발광 복합재료의

형태학적 특징 기반 미세구조 특성화 및 재구성

2019년 2월

서울대학교 대학원

기계항공공학부

김 영 환

Abstract

Morphological Feature-Based Microstructure Characterization and Reconstruction of Mechanoluminescent Composites through Particle Shape Library

YeongHwan Kim

Department of Mechanical and Aerospace Engineering

The Graduate School

Seoul National University

In this thesis, a new morphological feature-based microstructure characterization and reconstruction is developed for mechanoluminescent (ML) particulate composites. ML materials emit visible light proportional to the applied stresses. Therefore, ML materials have been studied for applications to stress sensor. It has also been known that the ML light intensity is proportional to the applied deviatoric stress in the particles. For the design of ML materials, it is critical to find out morphological features that can enhance the stress level within the particles. In the line of the research goal, morphological parameters that are most sensitive to

stress enhancement were determined through statistical correlation analyses between a set of morphological parameters and volume-averaged von Mises stress (VAS) of ML particles. To use actual morphological shapes of ML particles, an X-ray micro-computed tomography (CT) is used. To improve the image quality and segment each particle, a series of image processing algorithms such as Gaussian filter, median filter, and watershed algorithm are applied. Microstructure characterization was conducted based on thirteen morphological variables. Three-dimensional finite element analyses were conducted to obtain the VAS for each particle. The database that consisted of particles morphological parameters and VAS was generated and used to find the correlation between morphology and VAS. To perform this, the principal component analysis (PCA) is adopted to find out spectral components of the correlation matrix between morphological parameters and VAS. As a result of statistical analysis, the surface area and the shape index were found to be the most sensitive morphological parameters to the VAS and used for reconstruction of microstructures. A local dispersion of ML particles was reconstructed by the nearest neighbor distance (NND). One of the novel approaches adopted in this thesis was using a particle shape library that consists of voxelized 9,687 particle unit cells. The reconstruction was successfully accomplished by matching their probabilistic distributions with those of target parameters. An optimization algorithm, simulated annealing (SA) was adopted for matching distributions. A two-point correlation

function (TPCF) was used to verify the reconstructed microstructure. Also, the new analytical TPCF equation based on the morphological variables is generated with the reconstruction dataset. The algorithm proposed in this thesis has a salient advantage of computational efficiency and realistic microstructure reconstructed through the particle shape library. The combination of a dataset of reconstructed microstructure and novel data science is expected to be applied to the microstructure design of ML materials.

Keyword: Microstructure Characterization, Reconstruction, Shape Library, Morphological Analysis, 3D Finite Element Analysis, Mechanoluminescent Materials

Student Number: 2017-26274

Table of Contents

Table of Figures

Chapter 1. Introduction

Microstructure characterization and reconstruction (MCR) is a crucially important step toward materials by design for composite materials. In microstructure characterization and subsequent materials by design, consideration for realistic shapes and sizes of constituents allows for a rigorous characterization due to their close similarities to real morphological features — especially on the interface [1-3]. There are several traditional categories for statistical microstructure characterization. Statistical functions like two-point correlation function (TPCF), linear path function, two-point cluster function, etc. have been generally adopted to statistically characterize heterogeneous microstructures of composite materials [4-8]. Physical descriptors such as the nearest neighbor distance (NND), components size, etc. have also been applied for microstructure characterization [9-12].

The microstructure characterization often starts with microstructure images. To get the realistic geometries of constituents, an image data analysis with X-ray computed tomography (CT) is a powerful tool as a nondestructive method. However, getting the images through micro CT tests is time-consuming, costly, and skill-dependent. Therefore, it is not reasonable for applying micro CT tests to all regions of materials.

To overcome the limitations of the CT image acquisition, a reconstruction algorithm for generating statistically equivalent synthetic microstructure has been developed recently. Boustanabad et al. reviewed the recent volume of reconstruction research and categorized them into several cases [13]. The first case is a reconstruction using statistical functions like a two-point correlation function, linear path function, and two-point cluster function. Yeong et al. proposed an algorithm based on a probability function for 3D composite microstructure through 2D images [14]. Okabe et al. introduced a porous media reconstruction algorithm with multi-point correlation function [15]. In another stream of research, physical descriptors were applied as target parameters for reconstruction. Xu et al. proposed a reconstruction method that uses particles' morphology descriptors, for 3D particulate composites through 2D scanning electron microscopy (SEM) image [16]. Yang et al. used non-uniform rational B splines (NURBS) and particle shape library for generation particulate composites [17]. In the process of reconstruction, optimization is conducted to match the probability distribution both in probability function-based algorithms and physical descriptor-based algorithms. Generally, simulated annealing (SA) or generic algorithm (GA) is used for the optimization of probabilistic functions.

A state-of-the-art machine learning was also utilized for the reconstruction process. Hongyi Xu et al. used the machine learning to remove redundant descriptor to find essential variables for the microstructure design [18]. Sundararaghavan et al.

used a support vector machine to predict 3D reconstruction model from limited data of 2D image [19]. Bostanabad et al. applied supervised learning for stochastic microstructures to improve the efficiency of the reconstruction algorithm [20].

This thesis proposes a novel algorithm for MCR with particulate microstructure based on constituent morphologies. Micro CT images of mechanoluminescence (ML) composites were used as target microstructures. To improve the image quality and segment each particle, image processing algorithms such as a Gaussian filter, median filter, and watershed algorithm were applied. Microstructure characterization was conducted based on thirteen morphological variables. It is well known that the light intensity of the ML composite is proportional to deviatoric stresses in the particles. 3D finite element analyses were conducted to obtain volume averaged von Mises stresses (VAS) for each particle. The database that consisted of particles' morphological parameters and VAS was generated and used to find a correlation between morphology and VAS. To perform this, statistical methods like the principal component analysis (PCA) was applied. The most sensitive morphological parameters that are determined by the results were used to reconstruct statistically equivalent representative volume element (RVE).

Microstructure reconstruction is carried out within three steps: exporting particles, matching particles dispersion, and packing particles. A particle shape library based on voxel is applied to export particles for reflecting the realistic shape

of particles. A particle dispersion is matched by a nearest neighbor distance (NND) distribution. In the following section, microstructure characterization based on CT image is conducted. In Section 3, finite element analysis and statistical analysis are carried out. The two sections, Section 2 and Section 3, are based on author's paper, "*Effects of microstructure morphology on stress in mechanoluminescent particles: Micro CT image-based 3D finite element analyses*." published on "Composites Part A: Applied Science and Manufacturing" in November 2018 [21]. In Section 4, characterization and reconstruction algorithm that how to define particle dispersion with optimization using SA and how to packing particles is introduced with detail procedure. Finally, conclusions were summarized in Section 5 along with future research directions. Fig. 1 shows an overview of this thesis.

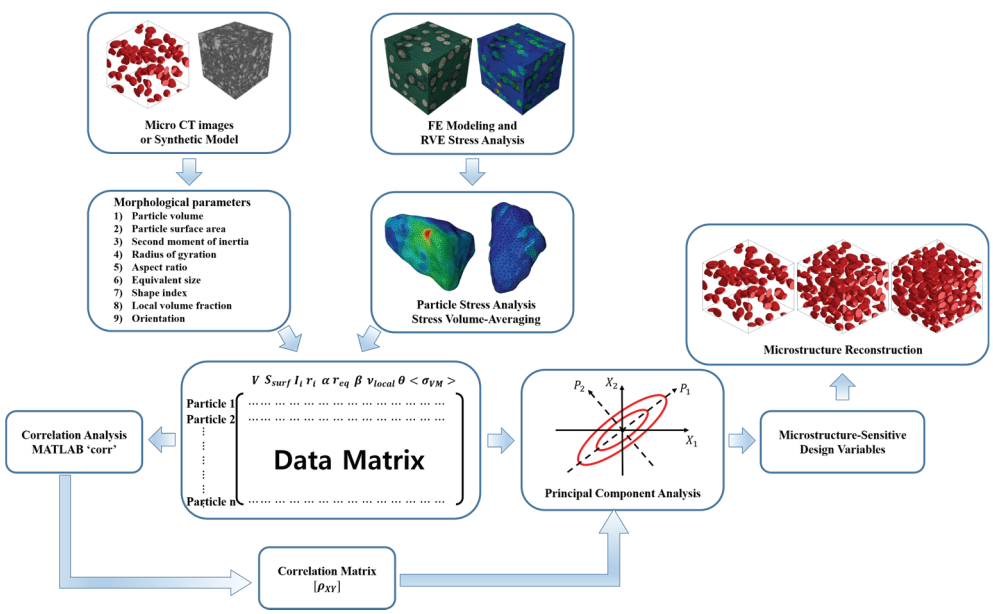


Fig. 1 The microstructure-sensitive design and reconstruction for ML particulate composites

Chapter 2. Microstructure Characterization

In this thesis, for the analysis of morphological descriptor and stress intensity, micro CT images of ML composites are used. A specimen is manufactured with ML powder and epoxy resin; the volume fraction of ML powder is 53.59%, and the mean size is 33 μm. The specimen is cylindrical whose diameter is 1 mm. The specimen is cut through heating it to make it accessible to process and polished using sandpapers.

2.1 Image Processing for Micro-Computed Tomography

The images of a specimen are taken with a 6C-beam line in Pohang Accelerator Laboratory (PAL, Pohang, South Korea) with the device setting shown as Fig. 2. A field of view (FOV) of the camera is $16.6\ \text{mm} \times 14.0\ \text{mm}$, and the pixel size is $6.5\ \mu\text{m}$. Considering that the ML particle mean size is $33\ \mu\text{m}$, an optical microscope of which magnification is ten times is applied. Ultimately a pixel size of the images is $0.65\ \mu\text{m}$. When the mean size of the particles is transferred as the number of pixels, approximately, the width of the particle in pixels is 50. Based on the number of pixels, the size of RVE is defined as $300 \times 300 \times 300$ (pixels), physically $195 \times 195 \times 195 \mu\text{m}$.

A set of image processing algorithms is applied to reduce the noise of images and to clarify interface between constituents. In this research, the image processing algorithms are generated in MATLAB code, and built-in functions in MATLAB image processing toolbox are used. First, a Gaussian filter, a median filter, and an erosion are applied to reduce the noise in the image. MATLAB built-in function 'imgaussfilt3' with smoothing kernel with a standard deviation of 0.5 (default) is applied as the Gaussian filter. The Median filter is used as built-in function 'medfilt3' with $3 \times 3 \times 3$ neighborhood size (default). Also, MATLAB built-in function 'imerode' that is set with the spherical shape structuring element object is applied to erode the images. Finally, 'imfill' is used for filling the holes in the particles. On the noise reduced images, a built-in function 'graythresh' based on Otsu's method is applied to define a proper threshold that discrete particles and a matrix phase. The image is binarized with the threshold; a matrix region is '0' and a particle region is '1'. From the result, particles are distinguished by a watershed algorithm, in MATLAB 'watershed'; as a result of the algorithm, the boundary of each particle is detected and generated. The boundary layer is contained to matrix region, so the particles do not share surfaces with other particles. The particle is defined as a set of voxels that share their six surfaces with other voxels. In Fig. 3 shows image processing algorithm except 'imfill'. In Fig. 3 (f), each particle that is separated with the watershed algorithm has a different color.

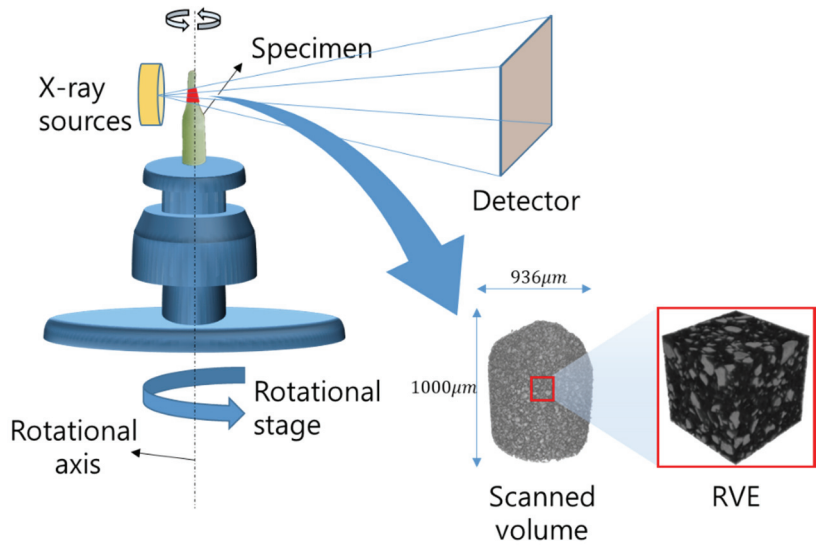


Fig. 2. Schematic process micro CT scanning and scanned volume and RVE element

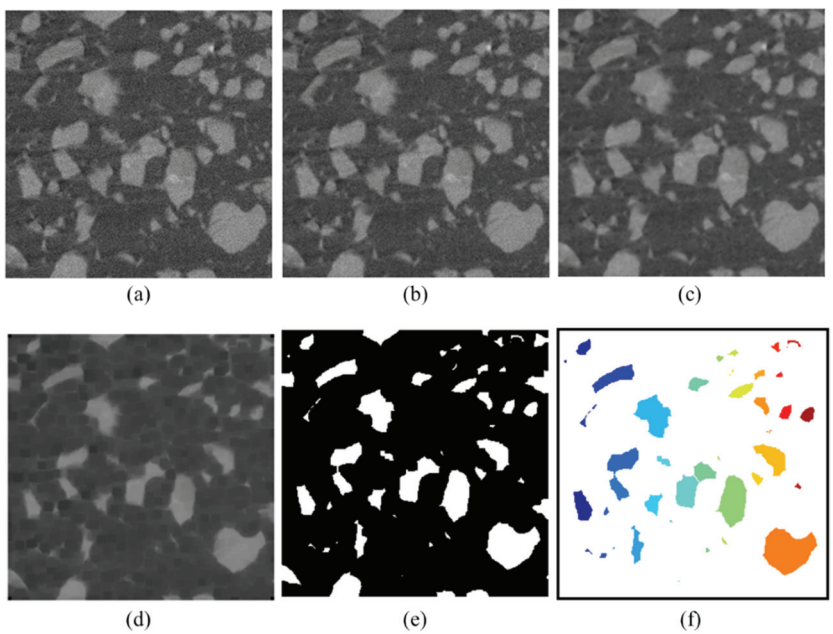


Fig. 3 Image processing flow (a) crop image, (b) Gaussian filter, (c) Median filter, (d) erode, (e) binarized, and (f) watershed algorithm

2.2 Quantitative Morphological Variables of ML Particles

Various morphological parameters are defined to investigate their individual effects on the particulate volume-averaged effective stresses. Following the image processing in Section 2.1, 3D voxelized image models are generated and used for calculating thirteen morphological variables based on voxel.

The volume of a particle (V) is calculated by multiplying the number of voxels (N_{voxel}) in the particle and a unit volume (V_0) of a voxel.

$$V = N_{voxel} V_0 \tag{1}$$

Then the surface area is calculated as follows

$$S_{surf} = N_{surf} S_o \tag{2}$$

where S_o is the area of one face of a voxel; N_{surf} is the number of surfaces that contact with matrix voxel. The second moment of inertia is calculated by

$$I_{ij} = V_0 \sum_{m=1}^{n} (x_i^m - \overline{X}_i)\left(x_j^m - \overline{X}_j\right) \text{ where } \overline{X}_i = \frac{1}{n}\sum_{m=1}^{n} x_i^m \tag{3}$$

where x_i^m is the coordinate of the center of the m-th voxel and $\overline{X}_i$ is the geometrical center of a particle in a Cartesian coordinate system (i=1,2, and 3). The principal second moments of inertia (I_1, I_2, and I_3) are eigenvalues of $[I_{ij}]$. Eigenvectors of $[I_{ij}]$ defines the principal directions. The principal radius of gyration is defined as

$$r_i = \sqrt{5I_i/V} \text{ where } r_1 > r_2 > r_3 \tag{4}$$

The orientation of a particle is defined as a projection angle of the first principal axis

onto the loading direction.

$$\theta = \cos^{-1}\left(\frac{\overrightarrow{\boldsymbol{p}_{I_1}} \cdot \vec{\boldsymbol{l}}}{\left\|\overrightarrow{\boldsymbol{p}_{I_1}}\right\|\left\|\vec{\boldsymbol{l}}\right\|}\right) \tag{5}$$

where $\overrightarrow{\boldsymbol{p}_{I_1}}$ is the vector for the first principal axis and $\vec{\boldsymbol{l}}$ is the vector to the loading direction as shown in Fig. 4 (a). The shape index (β) is also defined as

$$\beta = V/(S_{\text{surf}} r_1) \tag{6}$$

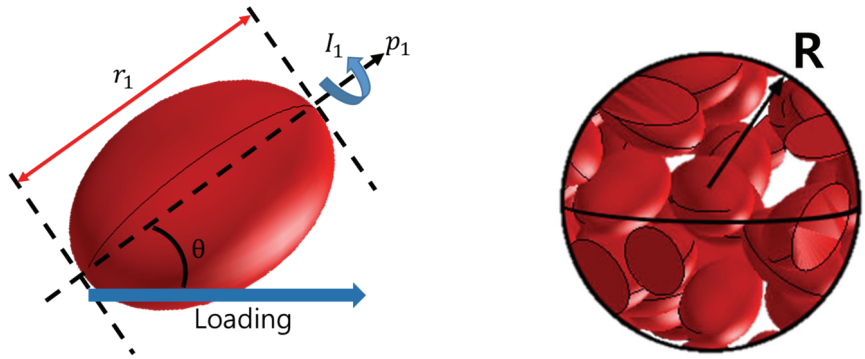
The aspect ratio (α) of a particle is defined as

$$\alpha = r_3/r_1 \tag{7}$$

Also, a local volume fraction per each particle is defined to measure the dispersion level of the particles. Setting a spherical local region of radius R as shown in Fig. 4 (b), the local volume fraction is computed as follows

$$v_{\text{local}} = \frac{\sum_{\text{i=1}}^{\text{n}} V_i}{\frac{4}{3}\pi R^3} \tag{8}$$

where V_i is a volume of particles included in the sphere. The volume of the spherical region was also computed using voxels.



(a) **(b)**

Fig. 4. (a) Orientation of an ellipsoidal particle and (b) clustering of particles within a spherical region

The equivalent size of a particle is defined as a radius of a sphere having the identical volume of a particle.

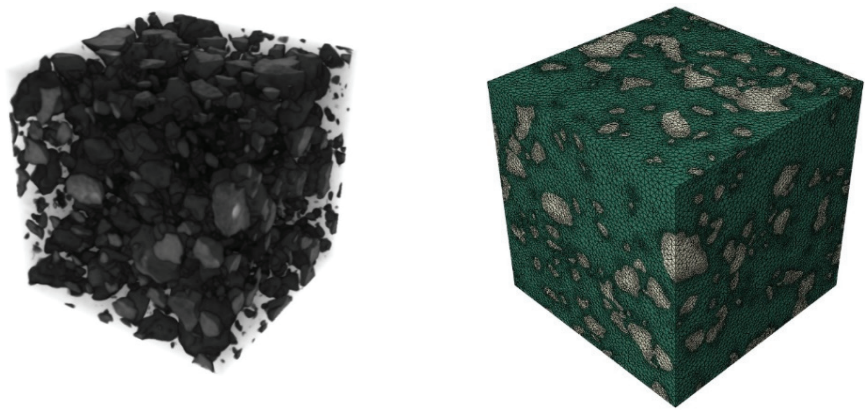
$$r_{\text{eq}} = \left(\frac{3}{4\pi} \times V\right)^{1/3} \tag{9}$$

Chapter 3. Microstructure Sensitivity Analysis

In this section, the most sensitive morphological parameters are determined among the variables that are defined in Section 2.2. Three-dimensional finite element analyses are conducted to find the stress contour on each particle. The finite element model is generated using Simpleware® based on the CT image set of ML specimen. For the same purpose in Section 2.1, a Gaussian filter, a Median filter, and a watershed algorithm are applied as a built-in function in Simpleware®.

3.1 Creation of Image-based FE Model

A three-dimensional FE model of the RVE was created as shown in Fig. 5. The total number of elements was 3,721,044. The number of elements for matrix was 2,652,980 and for particles was 1,068,064. Ten-node quadratic tetrahedron element (C3D10) in ABAQUS was used. The smallest element volume based on the RVE size that defined in Section 2, is 7.529×10^{-11} μm^3.



(a) (b)

Fig. 5 (a) Particle distribution from micro CT scan images, and (b) FE mesh of matrix + ML particles

The element set was defined for each particle in ABAQUS. The element sets are generated automatically using MATLAB in-house code. The ABAQUS input model has two element sets: particles and a matrix. The MATLAB code reads the input file and checks the connectivity of each element. Some elements that share a node are considered as one particle. The image process in Section 3 separates all the particle. Therefore, the number of sets was the same as the number of particles 971. Center coordinates of a particle were calculated by averaging coordinates of Gauss points of elements consisting of the particle. The center coordinates of voxels consisting of a particle were also computed by averaging the voxels' coordinates. Scaling the voxel space to the physical FE space, element sets could be paired with the particle by choosing minimum coordinate difference.

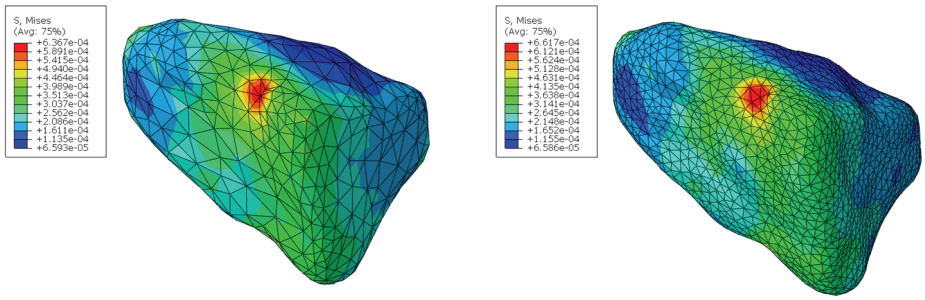
3.2 FE Mesh Sensitivity and Element Size Determination

This study focused on correlations of VAS and the morphological variables. The VAS of all particles were calculated through post-processing codes in python scripts. After FE analyses, VAS were computed as follows.

$$\sigma_{\text{eq}} = \frac{\sum_{i=1}^{N} \sigma_{VM\_i} \times V_i}{\sum_{i=1}^{N} V_i} = \frac{\sum_{i=1}^{N} \sigma_{VM\_i} \times \det(J_i)}{\sum_{i=1}^{N} \det(J_i)} \tag{10}$$

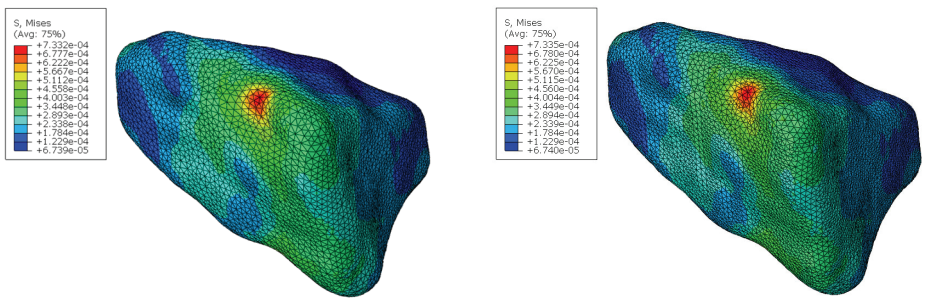
where σ_{VM_i} is the von Mises stress; N is the total number of Gauss points within a particle and V_i is the elemental volume which is identical to the determinant of the elemental Jacobian (J_i). The mesh-sensitivity study is performed to verify this result in this section.

A unit cell is built that contains a single particle constructed from X-ray CT images and applied uniaxial tensile stress to the unit cell. Ten different mesh densities were tested, and four of them are shown in Fig. 6 and Fig. 7. As expected, the maximum von Mises stress increased toward the exact solution. However, the variation of VAS values was minimal, which was max 0.3% difference for the particle shown in Fig. 6. Such insensitivity of VAS to the mesh size was attributed to decrease of the volume in the numerator of Eq. (10) and also confirmed the same result with different morphological shapes. Therefore, the FE mesh with the number of elements 7,364 in Fig. 6 (a) was selected in this paper. The fineness of Fig. 6 (d) deemed to be unnecessary considering a marginal improvement of accuracy from the sacrifice of computational costs.



(a) Number of elements=7,364; VAS=163.48 MPa

(b) Number of elements=18,641; VAS=163.11 MPa



(c) Number of elements=404,351; VAS=162.98 MPa

(d) Number of elements=783,652; VAS=162.99 MPa

Fig. 6 The number of elements and sensitivity of VAS and max von Mises stress to mesh density

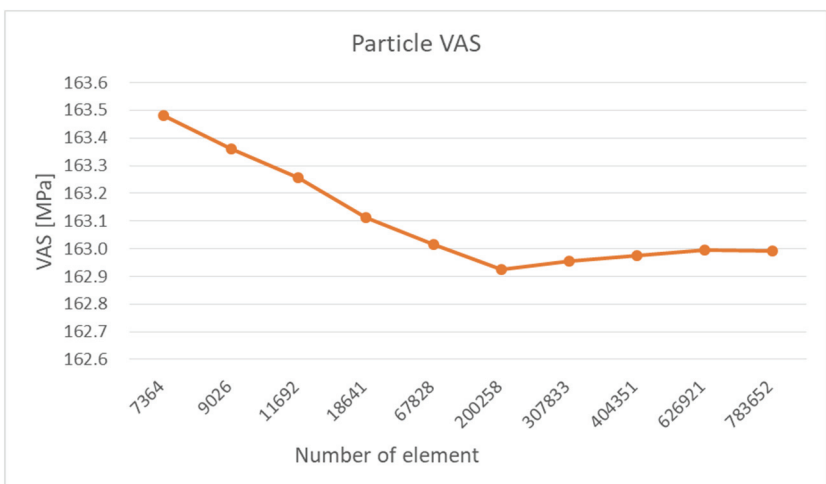
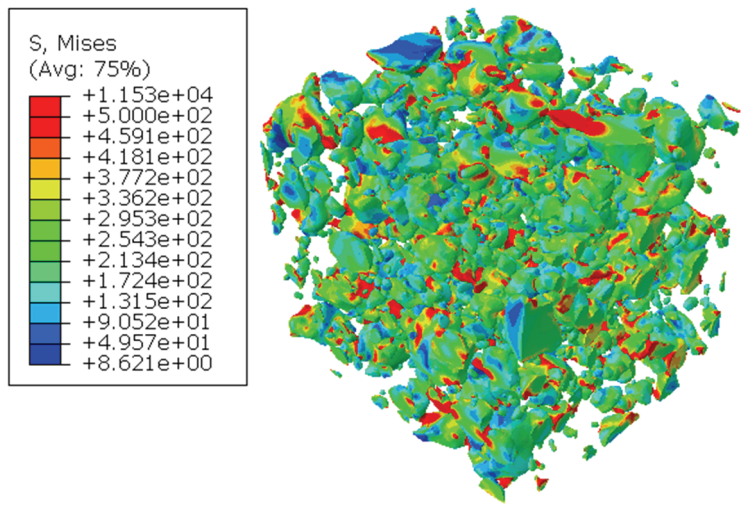


Fig. 7 The number of elements and sensitivity of VAS and max von Mises stress to mesh density in a graph

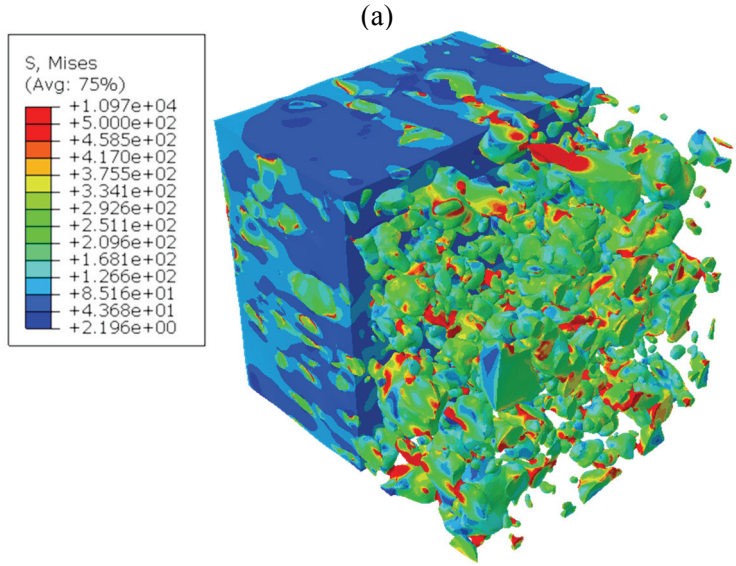
3.3 3D Finite Element Analysis for VAS of Particles

To obtain VAS in ML particles, linear elastic analysis using the 3D FE model of the RVE was conducted under the uniaxial stress (σ_{XX}=100MPa). Linear elastic properties of epoxy resin EpoxAcast 690 were E=3.94 GPa and Poisson ratio=0.3. Linear elastic properties of $SrAl_2O_4$:Eu^{2+},Dy^{3+} ML particles were E=102 GPa and Poisson ratio=0.23. This thesis assumed perfect bonding between ML particles and epoxy resin. With 100 cores (Intel® XEON, E5-2650v4) in parallel computation mode, it took about one hour. The total number of nodes of the 3D FE model was 5,079,399.

The contours of von Mises stresses are depicted in Fig. 8. As expected, stresses of the matrix were much lower than those of particles. Stress concentrations in the particles were observed from the simulation results.



(a)



(b)

Fig. 8 (a) von Mises stresses of particles and (b) half matrix and half particles from FE analysis

3.4 Correlation between VAS and morphological variables

Morphological variables of all particles were computed through image processing implemented in MATLAB codes. As described in Section 3.1, pairing the VAS of particles with the corresponding set of morphological variables was automated using MATLAB scripts. A quantitative statistical correlation analysis was conducted to find out the correlation between morphological variables and VAS. Both morphological variables and VAS were considered as random variables. A data matrix having a size of 971 by 14 was prepared. The number of rows (971) corresponds to the total number of particles. The number of columns (14) corresponds to the number of morphological variables (13) and VAS (1). This data matrix is also depicted in Fig. 1. MATLAB built-in function "corr" was applied to this data matrix in order to obtain the covariance matrix that is defined as

$$\sigma_{XY} = \boldsymbol{Cov}(X, Y) = \boldsymbol{E}[(X - \mu_X)(Y - \mu_Y)] \tag{11}$$

where X and Y are random variables; $\boldsymbol{E}[\cdot]$ is the expectation value; and μ_X, μ_Y are the mean of random variables. From the covariance matrix, the dimensionless coefficient of correlation can be obtained as

$$[\rho_{XY}] = \left[\frac{\boldsymbol{Cov}(X, Y)}{\sigma_X \sigma_Y}\right] \tag{12}$$

where σ_X and σ_Y are the standard deviations of random variable X and Y. Table 1 summarizes the coefficients of correlation of all morphological

variables with VAS (i.e.$\rho_{P,VAS}$, P=morphological variable). Fig. 9 indicates the correlation coefficients of all morphological variables. It is recalled that the morphological variables were divided into three groups: 1) size-related morphological variables (i.e., r_{eq}, V, S_{surf}, I_i, r_i with $i = 1, 2, 3$), 2) shape-related variables (i.e., θ, aspect ratio, and β), and 3) distribution-related variables (ν_{local}).

Table 1 Correlation coefficient of VAS and morphological variables for ML composite

1) Size-related variables					
r_{eq}	V	S_{surf}	I_1	I_2	I_3
0.2340	0.1098	0.1712	0.0725	0.0634	0.0466
r_1	r_2	r_3			
0.2730	0.2359	0.2139			
2) Shape-related variables			3) Distribution-related variables		
θ	α	β	ν_{local}		
-0.1007	-0.1246	-0.2681	-0.1178		

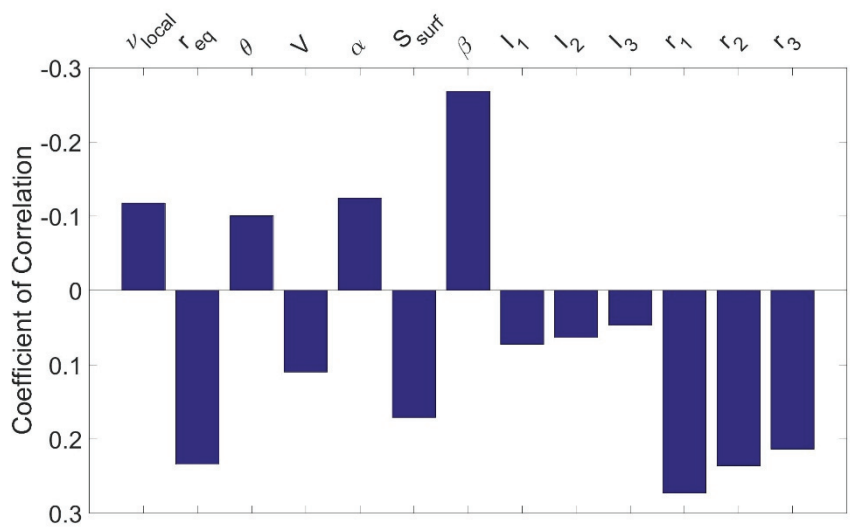
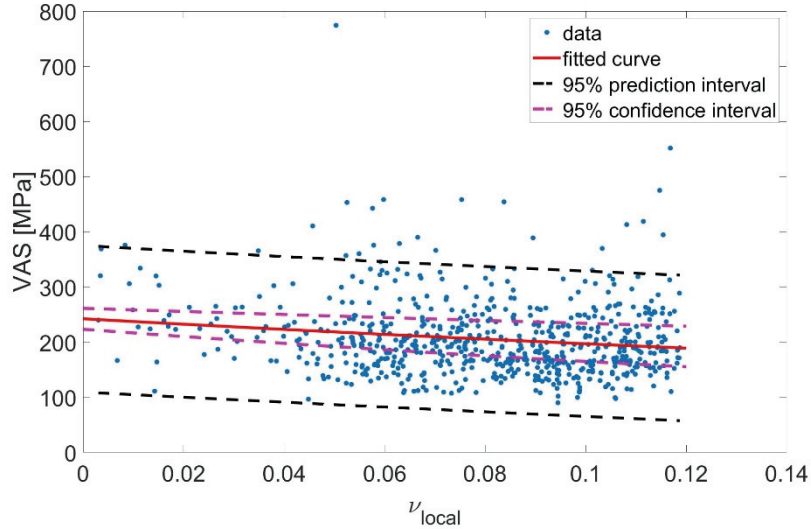


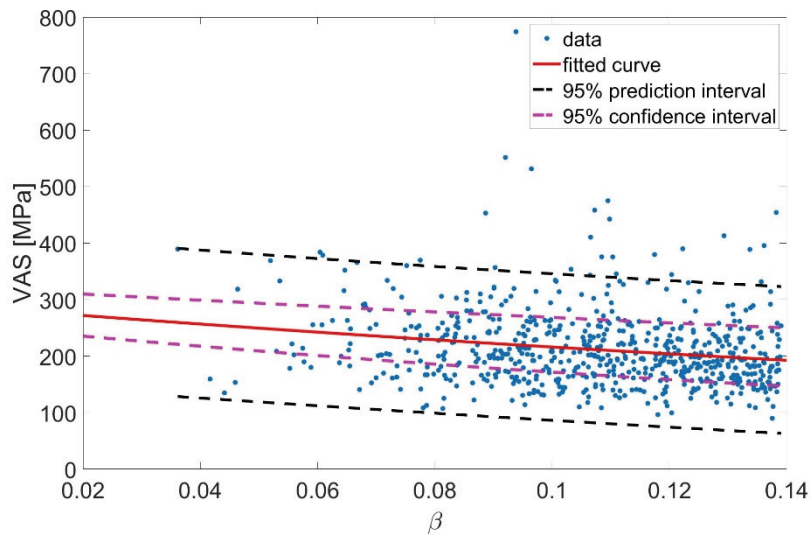
Fig. 9 Correlation coefficients between VAS and morphological variables

All of the size-related morphological variables have positive correlation coefficients with VAS whereas all of the shape and distribution-dependent morphological variables have negative correlation coefficients with VAS. This result implies that the stress is sensitive to the size, shape, and distribution (i.e., interaction) of the particles with varying degrees.

First, shape and distribution-dependent morphological variables versus VAS were plotted for all particles in Fig. 10 and Fig. 11. These variables seem to be uniformly distributed, for example, the shape index (β) is well distributed from 0.05 to 0.2. Among them, the shape index (β) has the highest correlation with the VAS. In the shape index range of 0.07-0.17, the highest VAS are populated. As shown in Fig. 12, the particle with a small shape index has higher VAS than the one with a large shape index. The contours in Fig. 12 do not represent the true stress of particles due to the perfect bonding assumption and mesh-dependency. However, as verified, VAS is not significantly affected by mesh-dependency and/or the possible singular stress concentration. This result is consistent with experimental observations in which platelet-like ML materials, that is, particles with small shape indices, showed higher sensitivity than rounded ML materials [22]. The aspect ratio is related to the shape index but less sensitive to the VAS than the shape index. Both the orientation and local volume fraction seem to have negligible correlation with VAS.

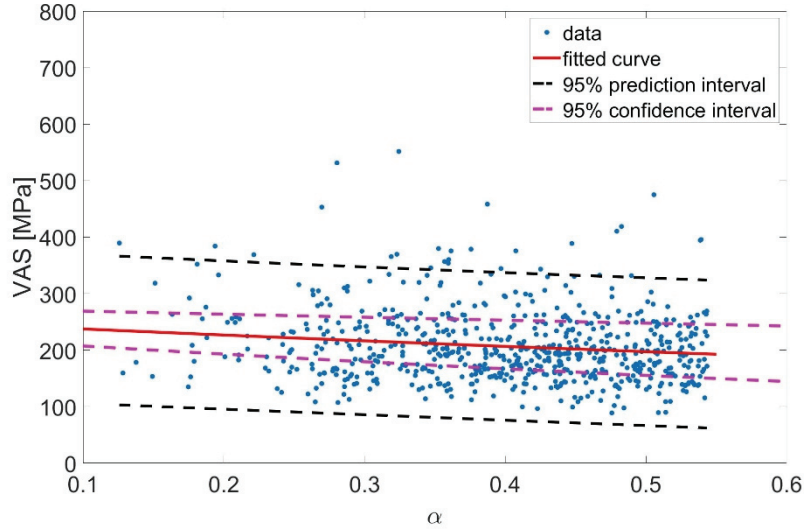


(a)

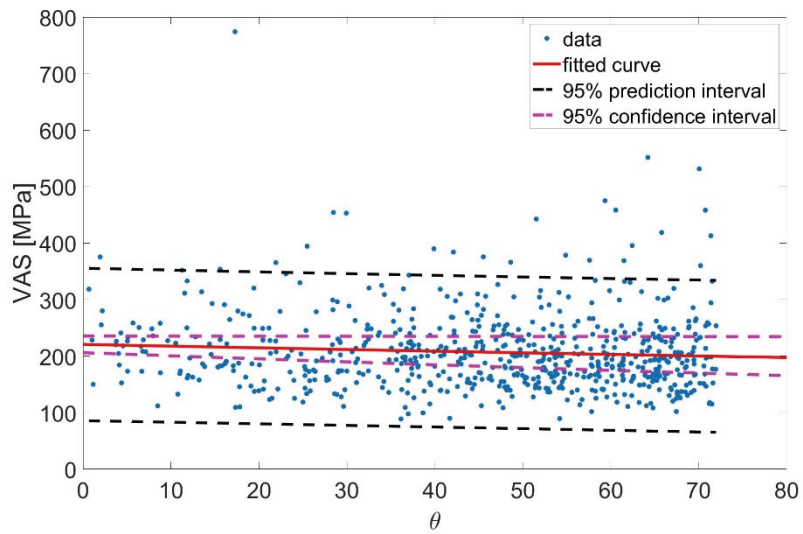


(b)

Fig. 10 VAS versus (a) local volume fraction and (b) shape index



(a)



(b)

Fig. 11 VAS versus (a) aspect ratio and (b) orientation angle

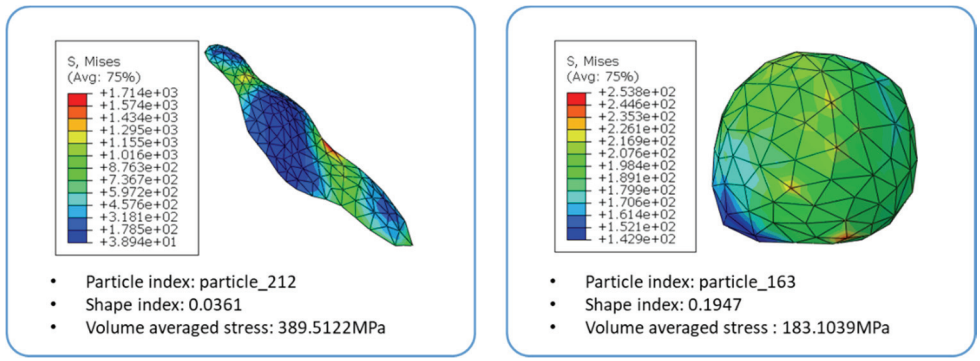
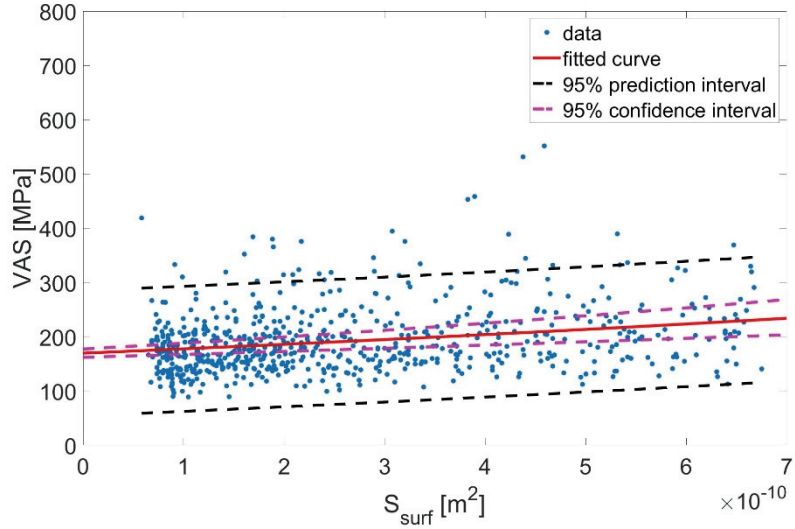
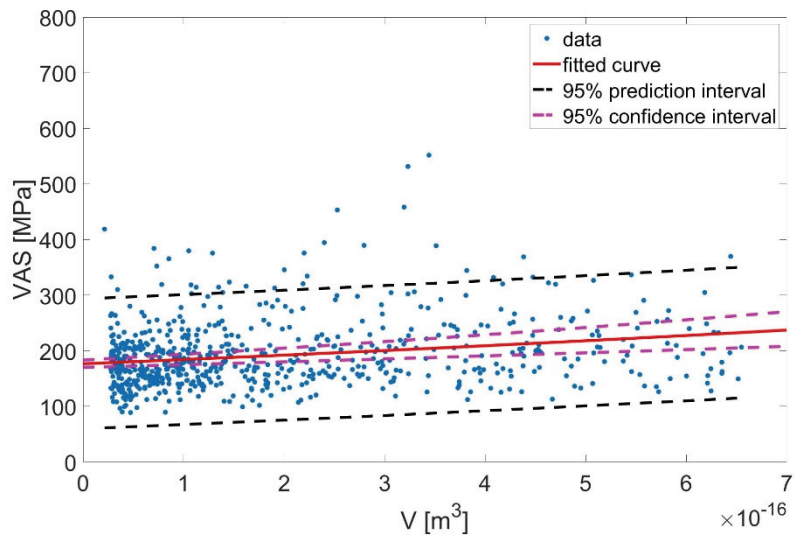


Fig. 12 Effect of shape index (β) on the particle shapes and stress

Second, VAS versus the size-dependent morphological variables for all particles were plotted from Fig. 13 to Fig. 17. Observing the distributions of the size-dependent variables, they are biased to a small size and the fraction of large size particles is negligible. Although all size-dependent variables have positive correlation coefficients with VAS, actually it is not true as shown in envelopes of from Fig. 13 to Fig. 17. The positive correlation coefficients were misled results because of the biased clouds in the small size band. Higher VAS occurs in small size particles for the variables (S_{surf}, V, r_{eq}, I_1, I_2, I_3, r_1, r_2, and r_3). Therefore, as the size of the particle decreases, the VAS tend to increase.

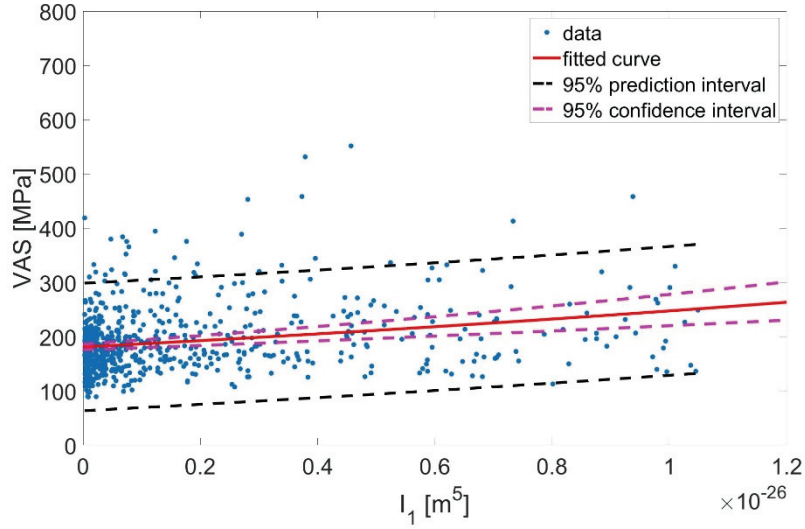


(a)

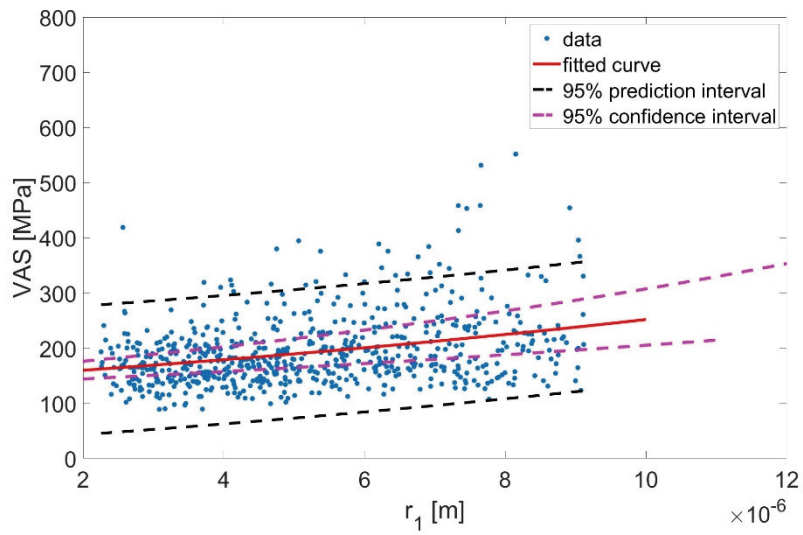


(b)

Fig. 13 VAS versus (a) surface area and (b) volume

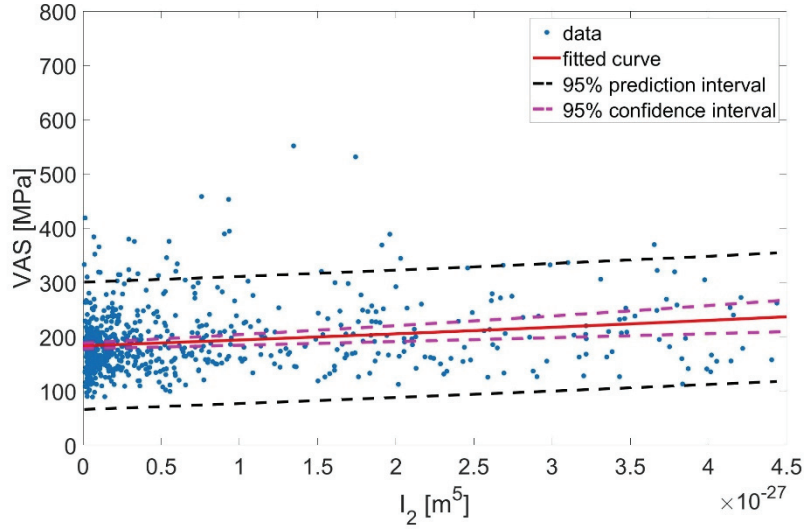


(a)

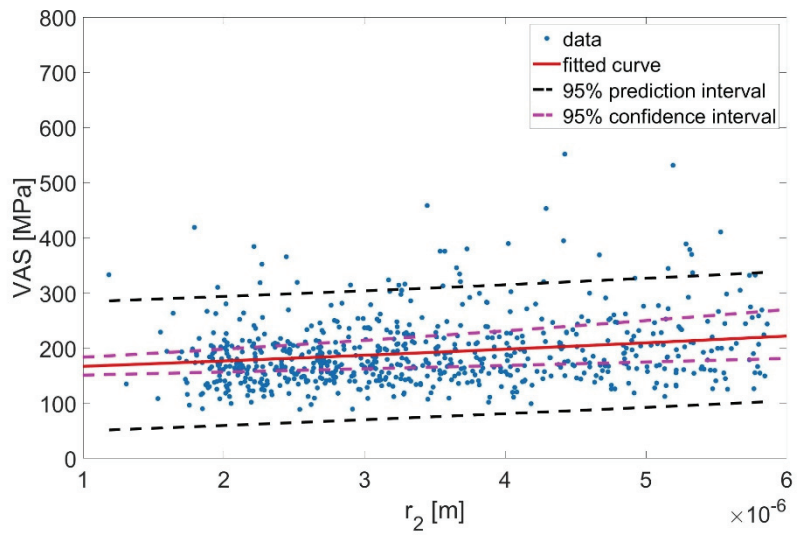


(b)

Fig. 14 VAS versus (a) 1st moment of inertia and (b) 1st radius of gyration

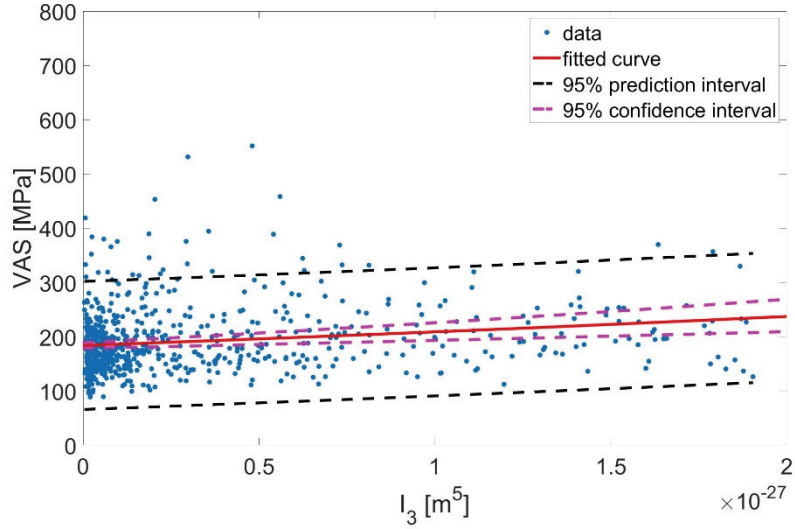


(a)

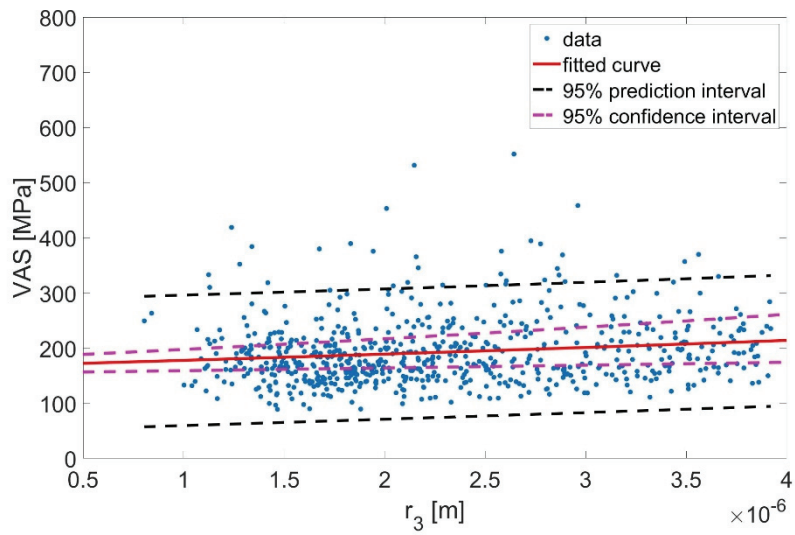


(b)

Fig. 15 VAS versus (a) 2nd moment of inertia and (b) 2nd radius of gyration



(a)



(b)

Fig. 16 VAS versus (a) 3rd moment of inertia and (b) 3rd radius of gyration

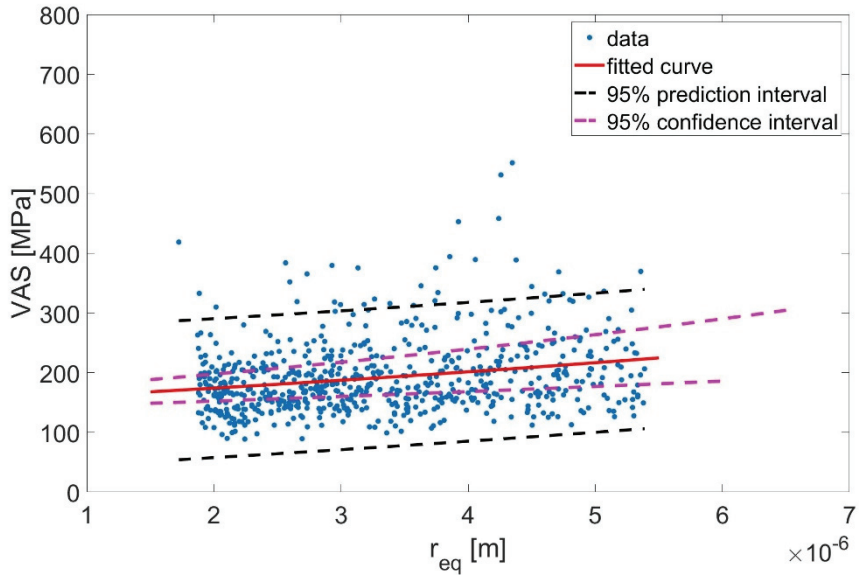


Fig. 17 VAS versus equivalent size

Fig. 18 shows a histogram of VAS. Most of VAS are scattered around 200 MPa. Based on this histogram, threshold VAS levels were determined to 300, 350, and 400 MPa to tracking the distributions of VAS per morphological variables.

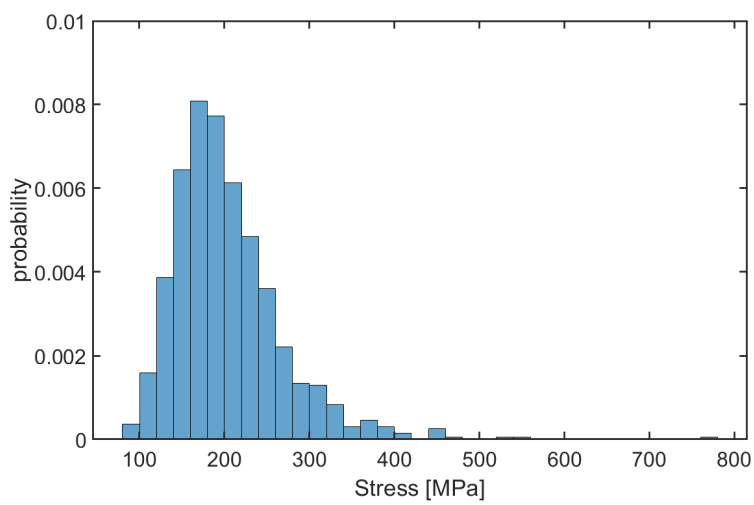
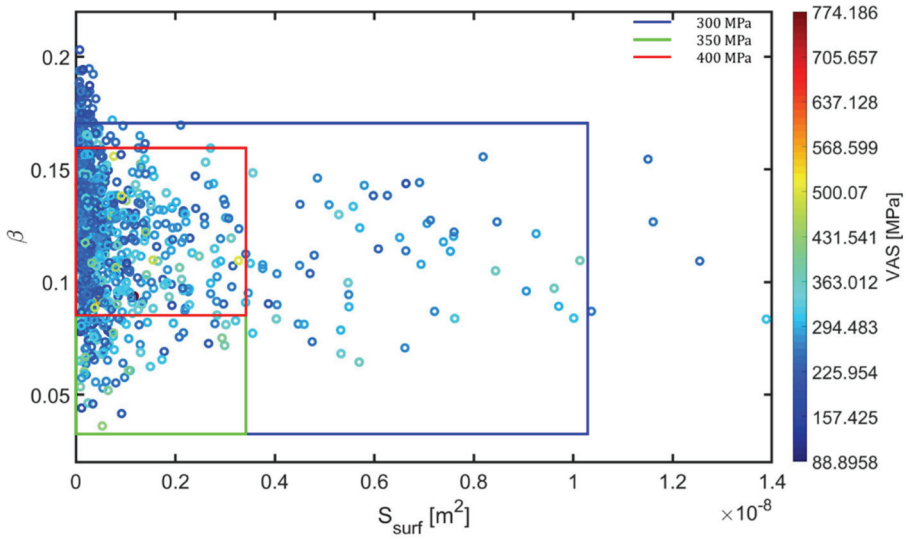


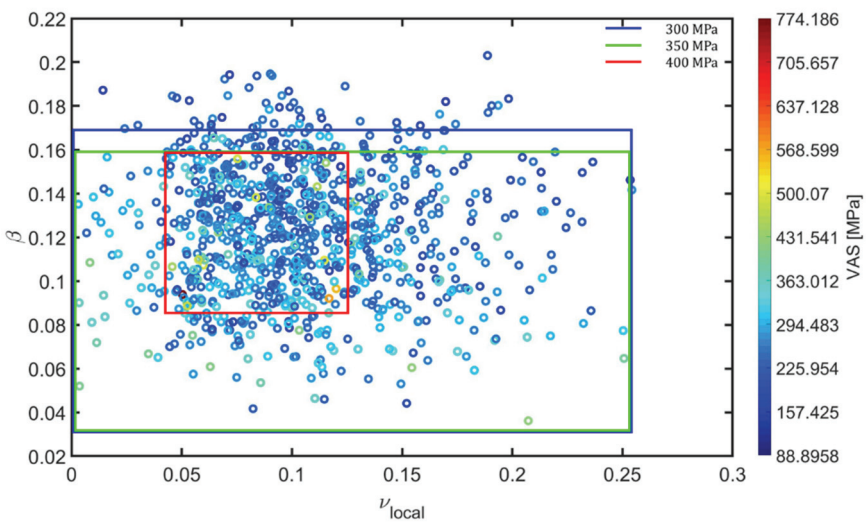
Fig. 18 Histogram of VAS of particles

Fig. 19 and Fig. 20 show joint distributions between two selected variables. Fig. 19 (a) shows joint distributions between the shape index and the surface area and Fig. 19 (b) between the shape index and local volume fraction. Particles having VAS more than 300, 350, and 400 MPa were enclosed by blue, green, and red rectangles respectively. According to Fig. 19 (a), particles having higher VAS tend to have a smaller surface area and have shape indices narrowed down between 0.085 and 0.16. However, there seems to be no correlation between the surface area and the shape index. According to Fig. 19 (b), particles having higher VAS tend to have local volume fraction narrowed down to a range of 0.04 and 0.127. The shape index also seems to have no correlation with the local volume. In these results, VAS can be significantly affected by the size, shape, and dispersion and they have no correlation with each other.

According to Fig. 20 (a), as the surface area decreases, the higher VAS develops. The surface area and the local volume fraction seem to have no correlation either. According to Fig. 20 (b), both surface area and r_1 are size-dependent morphological variables. Therefore, they show a high correlation with each other. These results also clearly indicate that small-sized particles have higher VAS than large-sized particles.

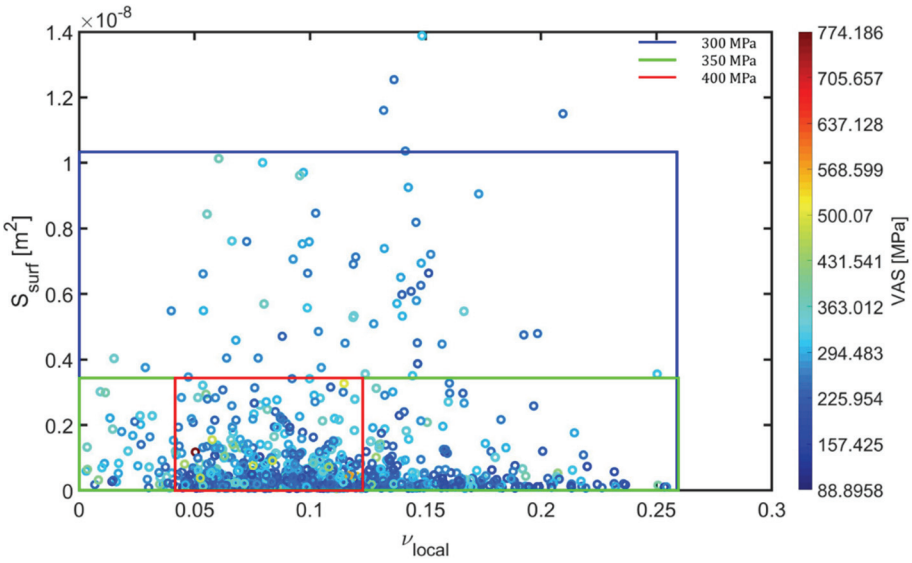


(a)

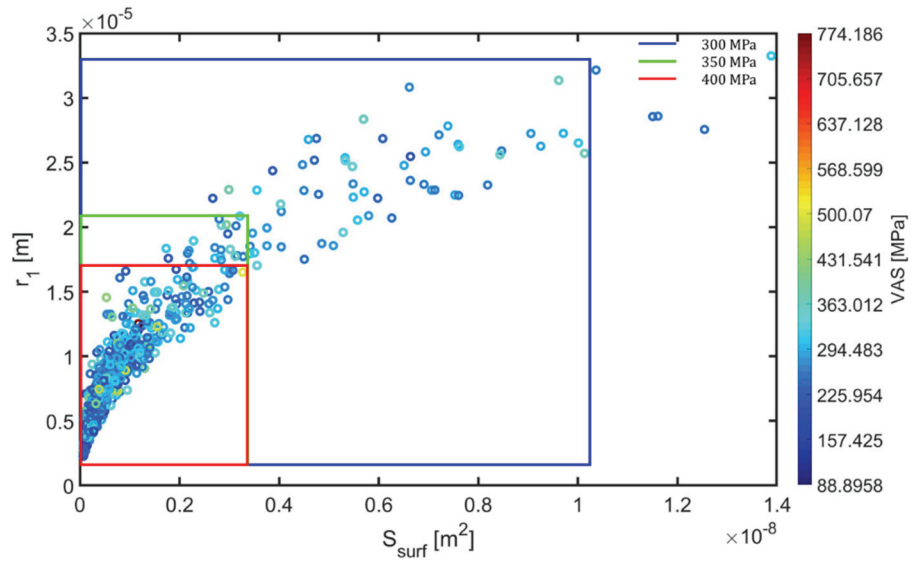


(b)

Fig. 19 (a) Joint distributions of shape index and surface area, (b) joint distributions of the shape index and local volume fraction with different threshold values



(a)

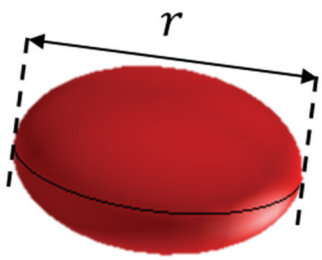


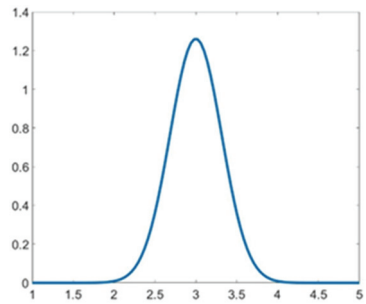
(b)

Fig. 20 (a) Joint distributions of shape index and surface area, (b) joint distributions of the shape index and local volume fraction with different threshold values

3.5. Design Parameters for Synthetic ML Particulate Composites

To ensure the negative correlation coefficients of the size-dependent variables with VAS, this thesis also generated nine synthetic microstructure RVEs having elliptical particles and conducted the same analysis. The synthetic RVEs were intentionally generated in a way that sizes of particles are well distributed. As summarized in Table 2, three-volume fractions and three different mean diameters in the first principal direction were assumed. By combinations of these, a total of nine synthetic microstructures were generated. Fig. 22 illustrates three synthetic microstructures filled with particles of the mean diameter 3 μm. Microstructure RVEs were generated so that they have periodic structures as shown in Fig. 22. Therefore, particles analyzed for morphological parameters include partially cut ellipsoids. This cut ellipsoid can naturally diversify morphological shapes of particles. Particle sizes were generated assuming the normal distribution with a variance of 0.1. Orientations of the particles were randomly generated following a uniform distribution. The size of RVEs depends on the particle size, with ascending order 15, 75, and 135 μm. All particles have an aspect ratio of 0.667.



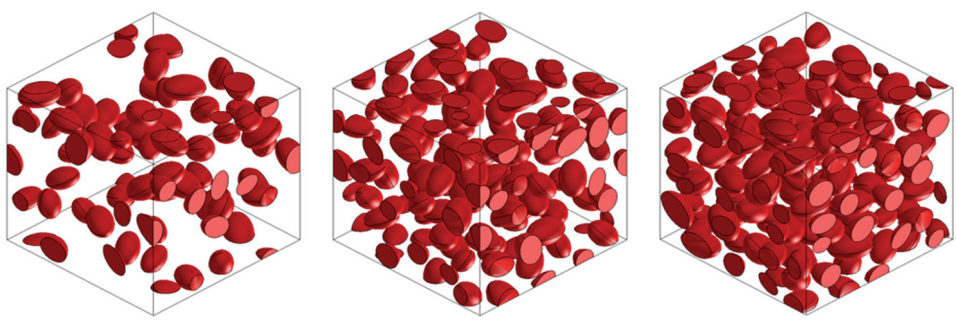


(a) (b)

Fig. 21 (a) Ellipsoidal particle and (b) normal distribution of particle size with a variance of 0.1;

Table 2 Nine synthetic microstructures of particulate composites

	Volume fraction		Mean Diameter		FF	F	M
A	0.10	FF	3 μm	A	A-FF	A-F	A-M
B	0.175	F	15 μm	B	B-FF	B-F	B-M
C	0.25	M	27 μm	C	C-FF	C-F	C-M

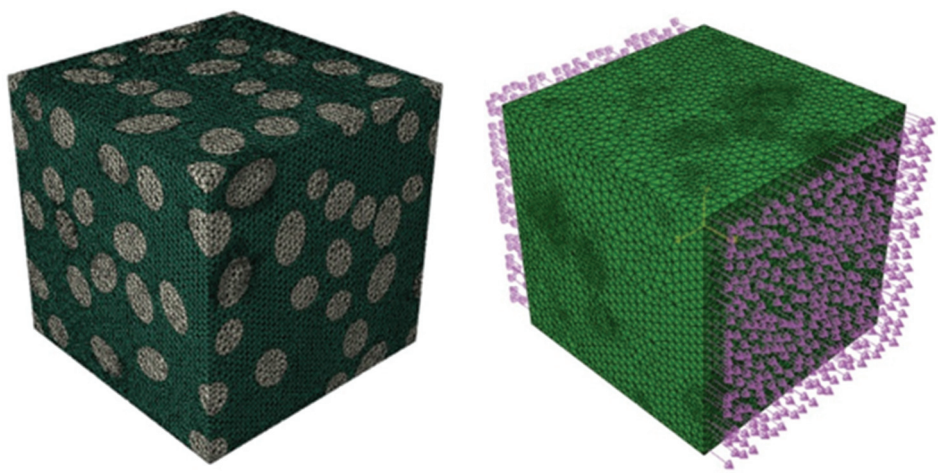


(a) (b) (c)

Fig. 22 Synthetic microstructure with three volume fractions, (a) 0.10, (b) 0.175, and (c) 0.25 and the same mean diameter 3 μm

The purpose of 3D finite element analyses is to compute volume-averaged effective stresses of all particles. 3D finite element models were developed for all nine synthetic RVE models. Linear tetrahedron finite element (C3D4) in ABAQUS

was used since it can easily comply with irregular shapes of particles. The numbers of elements for all synthetic RVEs are summarized in Table 3. Material properties for both matrix and particles were assumed to be isotropic linear elastic. The boundary condition and properties of the matrix and particles are same with Section 3.3. A 3D finite element mesh for one of the RVE (C-M) and boundary conditions are depicted in Fig. 23.



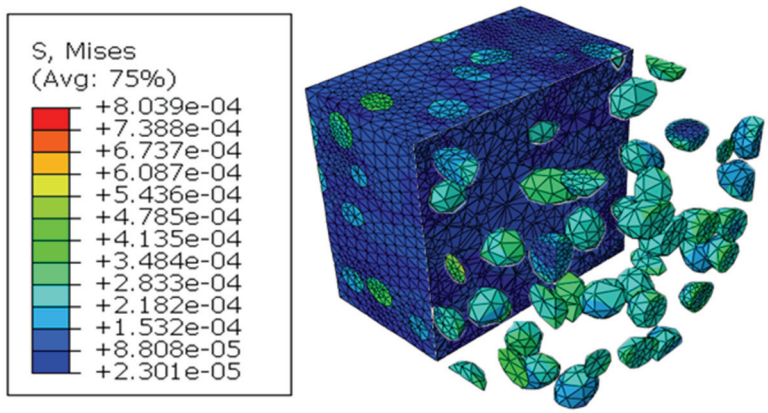
(a) (b)

Fig. 23 (a) 3D finite element mesh for C-M RVE and (b) Boundary condition

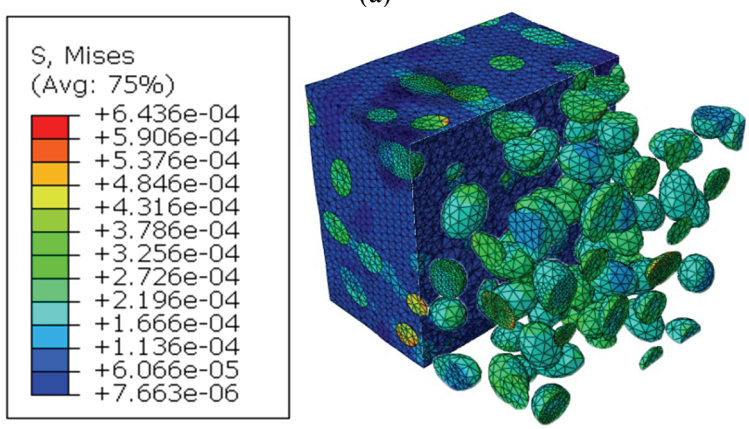
Table 3 Finite element model for each RVE

The number of elements		Volume fraction		
		0.1	0.175	0.25
Particle size	3 μm	579,820	59,485	123,591
	15 μm	53,165	151,906	133,774
	27 μm	148,589	70,914	440,081

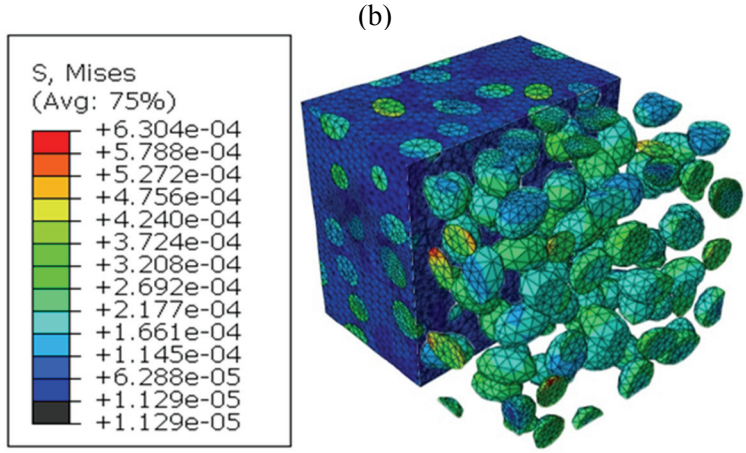
Fig. 24 shows the von Mises stress of the RVE that is consisted of 15μm particles for three different volume fractions.



(a)



(b)



(c)

Fig. 24 Finite element simulation results for volume fractions (a) 0.1, (b) 0.175, (c) 0.25

Pairing the volume-averaged von Mises stresses of individual particles with the corresponding set of morphological parameters was automated using MATLAB scripts. Elements for individual particle are a group to an element set, and by setting a tolerance to center coordinates of particles the element set and particles from 3D voxel images were paired.

Fig. 25 shows the correlation coefficients of morphological variables with VAS. Obviously, size-dependent variables appeared to have negative correlation coefficients with VAS except for r_1, and r_2.

These results show that the correlations between size-dependent parameters and VAS are generally negative whereas they are positive in real ML particles. These opposite results seem to be because of biased distributions of real ML particles to small size.

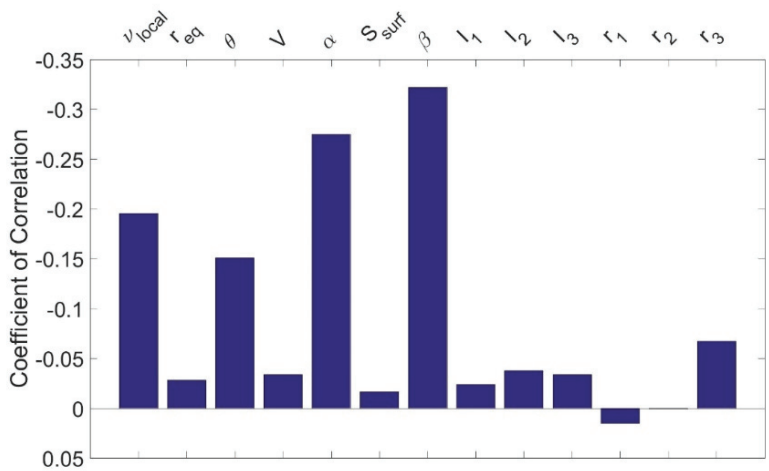


Fig. 25 Correlation of VAS and morphological variables for synthetic RVEs with elliptical particles

3.6 Principal Component Analysis

In Section 3, morphological variables are classified into 1) size-dependent, 2) shape-dependent and 3) distribution-dependent groups. Morphological variables in each of the groups are correlated. Therefore, considering all of them is redundant. For reducing these redundant variables and finding uncorrelated variables, the principal component analysis (PCA) is widely used [23]. In this paper, PCA is conducted on the 14-by-14 correlation coefficient matrix as an eigenvalue problem. The last column of the correlation coefficient matrix contains correlation (i.e., $\rho_{P,VAS}$, P: morphological variable) between VAS and all morphological variables.

$$[\rho_{xy}]\boldsymbol{\phi} = \boldsymbol{\phi}\lambda \quad \textbf{(13)}$$

where $[\rho_{xy}]$ is the correlation coefficient matrix; $\boldsymbol{\phi}$ is the eigenvector and λ is the eigenvalue. If the transpose of the normalized eigenvector is post-multiplied on both sides, $[\rho_{xy}]$ can be expressed as

$$[\rho_{xy}] = \boldsymbol{\phi}\lambda\boldsymbol{\phi}^{\mathrm{T}}. \quad \textbf{(14)}$$

The diagonal terms of $[\rho_{xy}]$ matrix are the coefficients of autocorrelation for morphological variables. The off-diagonal terms of $[\rho_{xy}]$ are the coefficients of cross-correlation between morphological variables. Therefore, the terms are '1'. Eq. (14) is decomposed as a summation of all components.

$$[\rho_{xy}] = \sum_{\mathrm{i}=1}^{\mathrm{n}} \phi_i \lambda_i \phi_i^T \quad \textbf{(15)}$$

Only three largest eigenvalues are considered. Therefore, Eq. (15) is reduced as follows

$$\left[\rho_{xy}\right] \cong \phi_1 \lambda_1 \phi_1^T + \phi_2 \lambda_2 \phi_2^T + \phi_3 \lambda_3 \phi_3^T \quad \textbf{(16)}$$

In Eq. (16), each term on the right-hand side is in a matrix form that has the same size as the $\left[\rho_{xy}\right]$ matrix. The diagonal terms in each term of Eq. (16) are the principal components of $\left[\rho_{xy}\right]$ matrix. For example, each diagonal term of $\phi_1 \lambda_1 \phi_1^T$ is the first principal component of each morphological variable. In the same way, the second and third principal components can be obtained from $\phi_2 \lambda_2 \phi_2^T$ and $\phi_3 \lambda_3 \phi_3^T$, respectively. Table 4 lists eigenvalues of PC values and their percentage. The first three PC values are dominants up to about 80%.

Table 4 Eigenvalues and variance percentage of all PC values

Component	**PC1**	**PC2**	**PC3**	**PC4**
Eigen Value	7.7860	2.0775	1.2751	1.0038
Variance [%]	55.6144	14.8395	9.1076	7.1697
Component	**PC5**	**PC6**	**PC7**	**PC8**
Eigen Value	0.8357	0.7301	0.1155	0.0667
Variance [%]	5.9690	5.2152	0.8246	0.4765
Component	**PC9**	**PC10**	**PC11**	**PC12**
Eigen Value	0.0496	0.0247	0.0177	0.0110
Variance [%]	0.3542	0.1766	0.1267	0.0784
Component	**PC13**	**PC14**		
Eigen Value	0.0057	0.0010		
Variance [%]	0.0404	0.0072		

The principal components of correlation coefficients between morphological variables and VAS (i.e. $\rho_{P,VAS}$) were plotted along the axis of each principal component in Fig. 26. In Fig. 26, the component that is more aligned to a principal axis has a higher contribution to that principal component. All of the size-dependent variables are in PC1 direction. It implies that the angle between vectors represents their correlation. The vectors representing shape-dependent variables are in PC2 direction, which is uncorrelated with size-dependent variables in PC1 direction. The longer vector is, the more variability of the variable is in the

corresponding PC direction. The results of PCA reduce the number of morphological variables to three, that is, the surface area (S_{surf}), the local volume fraction (v_{local}) and the shape index (β). The shape-dependent variables are aligned to PC2 axis, showing very high correlation with each other. The local volume fraction (v_{local}) is aligned with PC3. Therefore, the three variables are considered the most microstructure-sensitive design variables, which are statistically independent of each other. Table 5 shows the principal component values ($\phi_1\lambda_1\phi_1^T$, $\phi_2\lambda_2\phi_2^T$, and $\phi_3\lambda_3\phi_3^T$) of the $\rho_{P,VAS}$ for S_{surf}, v_{local}, and β. As described in Section 3.4, the misled correlations of size-dependent variables with VAS due to the small number of large-sized particles made positive PC1 in Fig. 26.

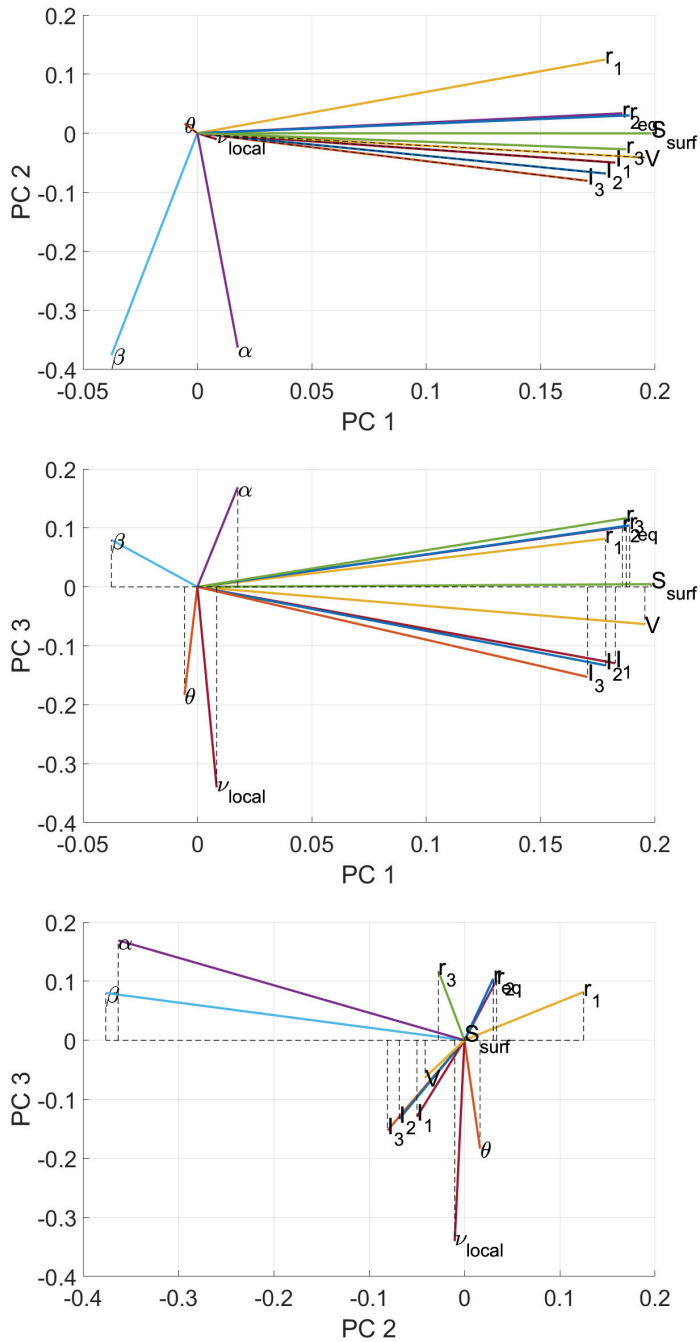


Fig. 26 Principal component analysis results for ML composite

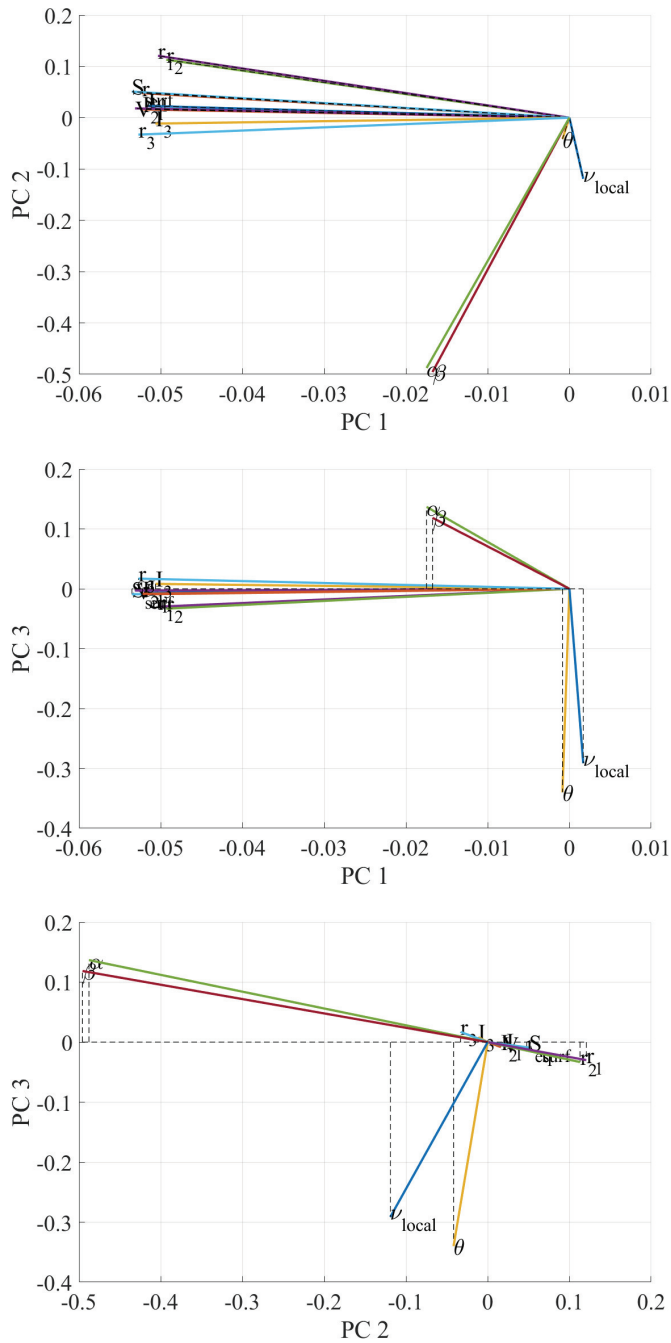


Fig. 27 Principal component analysis results of synthetic microstructure RVE

Fig. 27 shows PC analysis results from the nine synthetic microstructure RVEs. They also showed similar characteristics of PC values except for negative PC values of the size-dependent variables.

Table 5 Principal component values for the morphological variables of real ML particles

Morphological variables	PC1	PC2	PC3
S_{surf} - VAS	0.198214	-0.000176	0.0045790
β - VAS	-0.037562	-0.375997	0.07993369
v_{local} - VAS	0.008429	-0.0101719	-0.3401004

Chapter 4. Microstructure Reconstruction

In this section, a novel algorithm that reconstructs statistically equivalent microstructure based on morphological descriptors is described. In the previous section, the surface area and shape index of particles are the most sensitive variables to volume averaged effective stress. Therefore, the reconstruction is conducted based on the two parameters. There are three steps for generating statistically equivalent RVE. First of all, particles are exported from the shape library that has the same distributions of the target particle surface area and shape index. After sampling particles, a particle center map that is based on NND distribution is made. Finally, the particles that are exported from the particle library are laid at the position that is defined by the particle center map. The flow chart of reconstruction algorithm is shown in Fig. 28. In this section, it generates a statistically equivalent RVE model through reconstruction algorithm. The target RVE size and reconstructed RVEs size are $150 \times 150 \times 150$ (pixels). There are five regions of target RVE in the same specimen. For each of the target RVE models, ten reconstructed models are generated. Therefore, the number of a total reconstructed model is fifty.

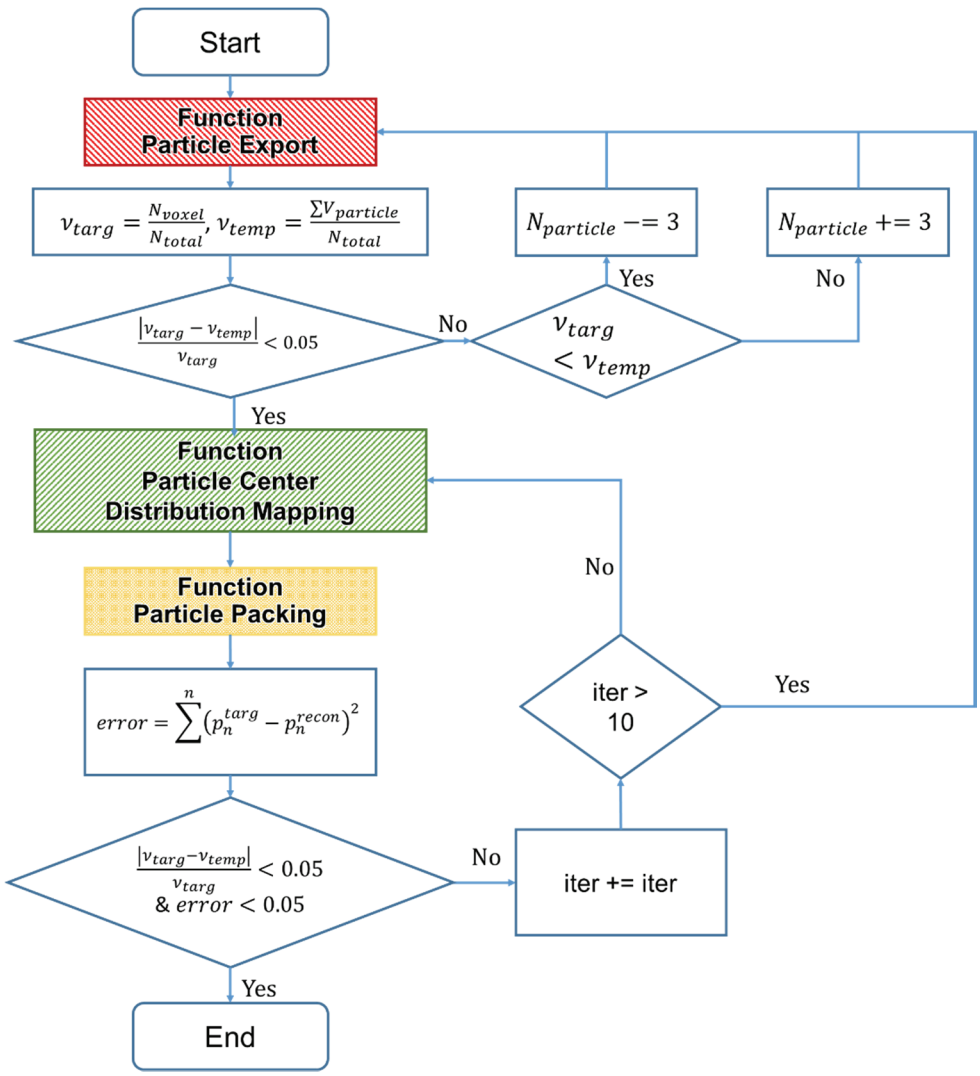


Fig. 28 Flow chart of a microstructure reconstruction algorithm

Algorithm 1. Reconstruction

```
// main program for reconstruction
errorVF ← relative error of the volume fraction
errorPart ← particle parameter distribution error calculated with least square method
while errorVF>0.05 && errorPart > 0.05
    while errorPart > 0.05
        ShapeMatch(numParticle, target, particleLib) ← particle export, Algorithm 2
        if targVF < reconVF
            numParticle -= 3
        else if targVF > reconVF
            numParticle +=3
        end if
    end while
    DisperMatch(imageSize, numParticle, target) ← dispersion matching, Algorithm 3
    particlePacking ← packing exported particle at the particle center position, except
                      particle overlapping, Algorithm 4
end while
```

4.1 Particle Sampling from Particle Shape Library

As observed in the real microstructure of ML composites, the ML particles have arbitrary shapes; hence it is not proper to assume the shape of particles as either ellipsoidal or spherical shape. To accommodate the arbitrary shapes of ML particles, the particle shape library is constructed from which particles are sampled. The library consists of various shapes of the particle unit cell models. The particle unit cell model is a three-dimensional matrix that contains a voxelized particle. The unit cell models of the particle are shown in Fig. 29.

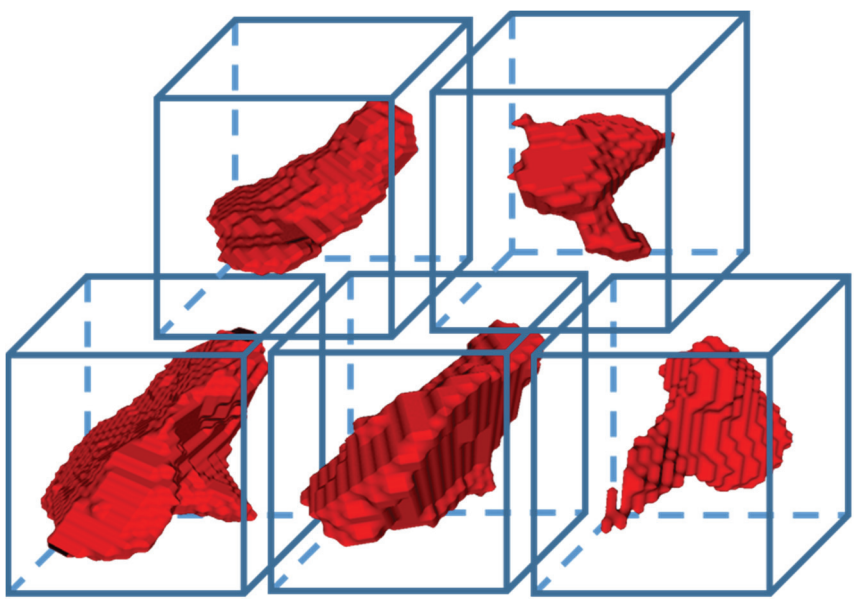


Fig. 29 Voxelized particle unit cell models

The particle shape library is constructed with stacked images which size is $780 \times 822 \times 1100$ (pixels). The image processing proceeds with the same operations described in Section 2.1. However, for reflecting the real particles' shapes,

particles cut with boundaries have to be removed because they are in unrealistic artificial shape. MATLAB built-in function 'imclearborder' is used to remove particles that are cut by the boundaries. Finally, the particle shape library contains the information of 9,687 particles.

The procedure for particle sampling is to match the cumulated distribution function (CDF) histogram values between targeted and reconstructed ones. A histogram of the surface area (or shape index) distribution should be made with proper bins. If the number of bins is too small, the convergence of reconstructed distribution is fast, but the accuracy of the distribution is low. Inversely, too many bins make the high accuracy, but it may cause convergence issue. An initial distribution for the reconstructed model is randomly sampled in the range of target bins. At the sampled values, the value of one interval is decreased by one and add one in another. This process is conducted with an optimization algorithm, simulated annealing (SA), and the objective function is a minimization error — the difference between targeted and reconstructed CDF histogram.

The error is calculated as a least square method with two histograms, and it is defined as

$$error = \sum^{\mathrm{n}} \left(p_n^{targ} - p_n^{recon}\right)^2 \tag{17}$$

where p_n is the cumulated probability of n-th bin, and the superscripts (*targ*, and *recon*) mean the target and reconstruction. The SA algorithm is conducted following

procedures. When the sampled value is modified (for convenience, this is called a new step), the *error* also changed. If the new *error* is decreased, that means the similarity between target and reconstruction distribution is increased. Then, the new step is accepted. However, when the new *error* is increased, the new step will be admitted following probability,

$$p^{accept} = \exp\left(\frac{error^{new} - error^{old}}{T}\right) \tag{18}$$

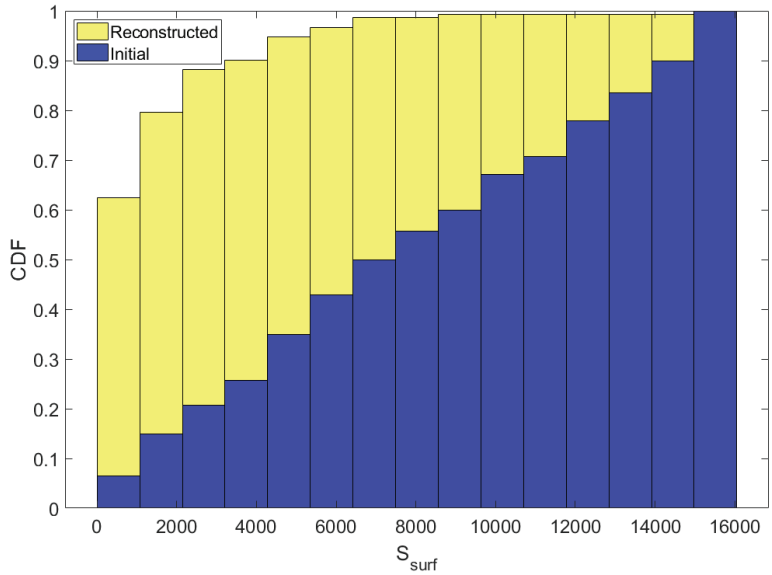
where T is the control value that is decreased as the step progress like Eq. (19)

$$T = \frac{1.0 \times 10^{-5}}{t} \tag{19}$$

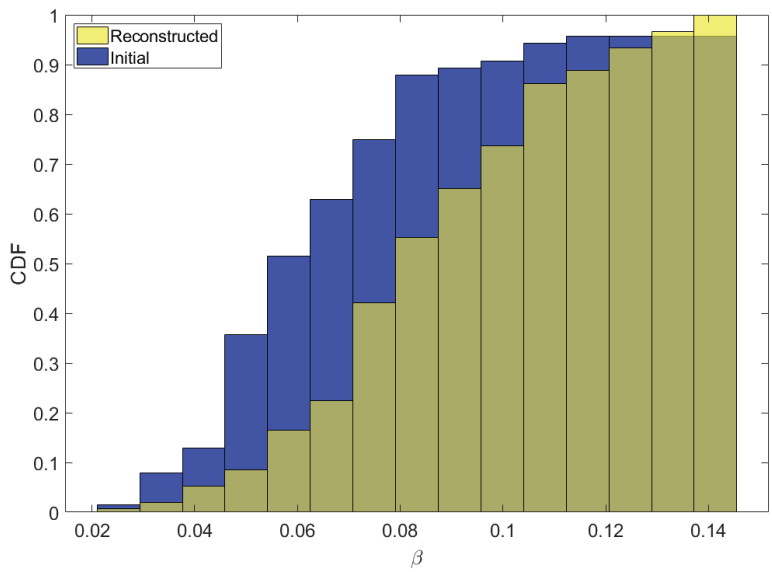
where t is the number of accepted steps. The SA algorithm accepts the step that the *error* is increased in the beginning, but as the number of iterations increases, the bad change is ignored. As iteration proceeds, the distributions get converged to the targeted CDF from Fig. 30 to Fig. 31. Fig. 32 shows the error reduction of surface area and shape index during iteration.

The number of bins for the surface area and shape index is determined to be fifteen considering a convergence of the algorithm and the total size of intervals. In the histogram, the particles in the range of the maximum value and the minimum value of each interval are searched and randomly extracted. In the histogram, each bin has minimum and maximum values, so the probability of each bin means the likelihood of existing particles between the maximum and minimum values. When

the number of total particles is defined, the number of particles in each bin is based on its probability. Following the number of particles, the particle unit cells are randomly exported in the boundary of each bin. In this thesis, the reconstructed model satisfied two morphological parameters. Therefore, each particle is exported in the intersection of them like Fig. 33. Moreover, the distribution of all the particle in the particle library is shown in Fig. 34. The flow chart of particle export algorithm is shown in Fig. 35.



(a)



(b)

Fig. 30 Histogram of (a) surface area and (b) shape index distribution of initial model and reconstructed model

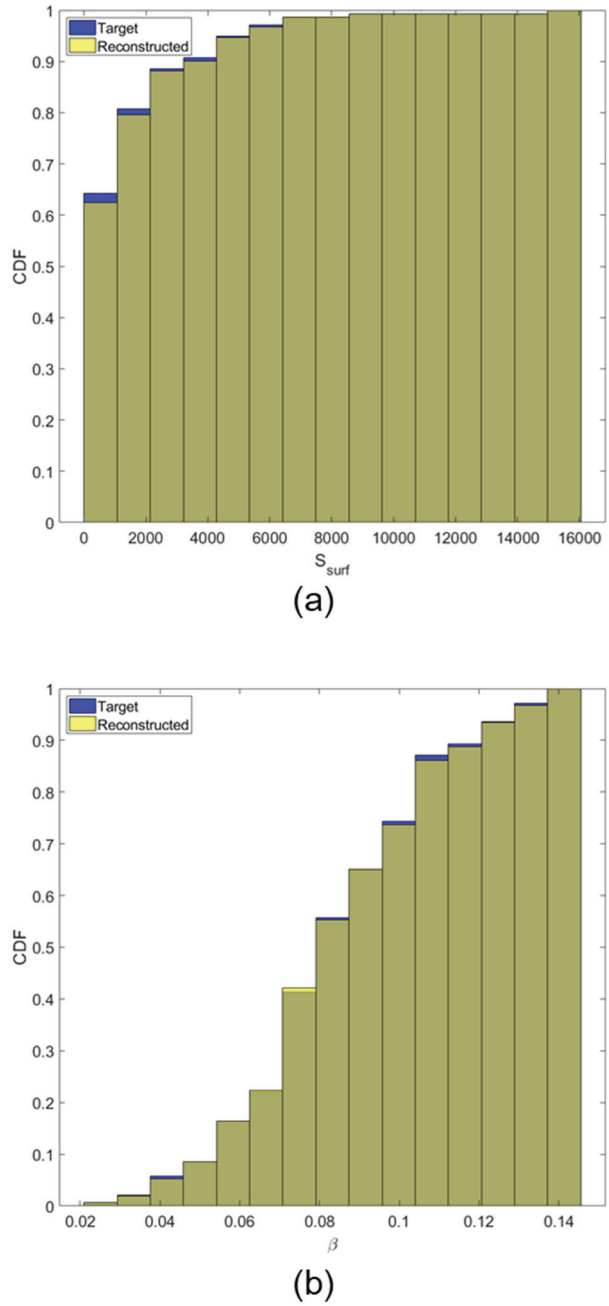
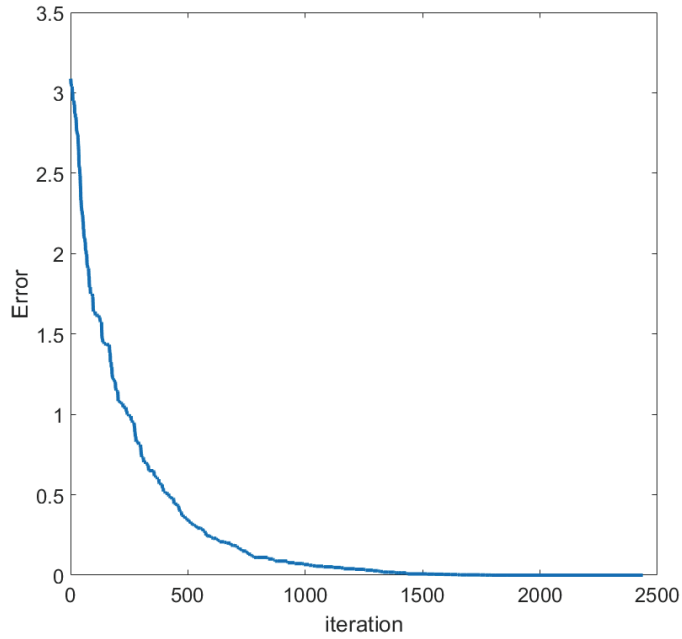
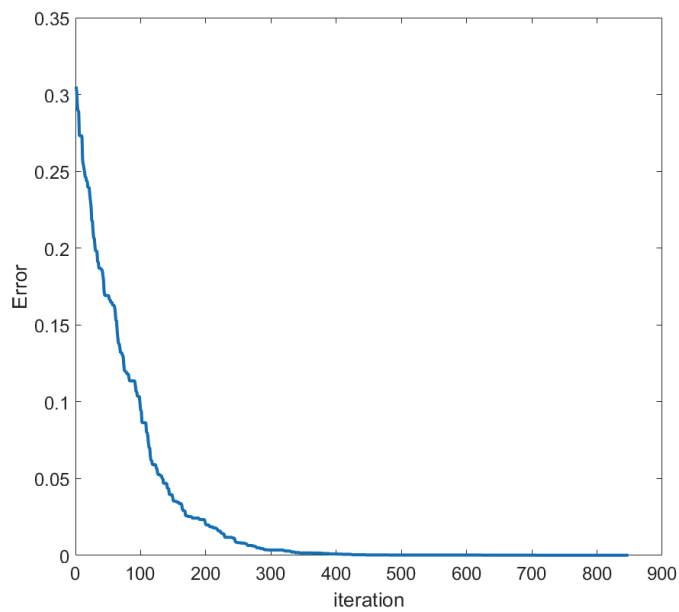


Fig. 31 Histograms of (a) surface area and (b) shape index distribution of target model and reconstructed model



(a)



(b)

Fig. 32 Error reduction of (a) surface area and (b) shape index during iteration

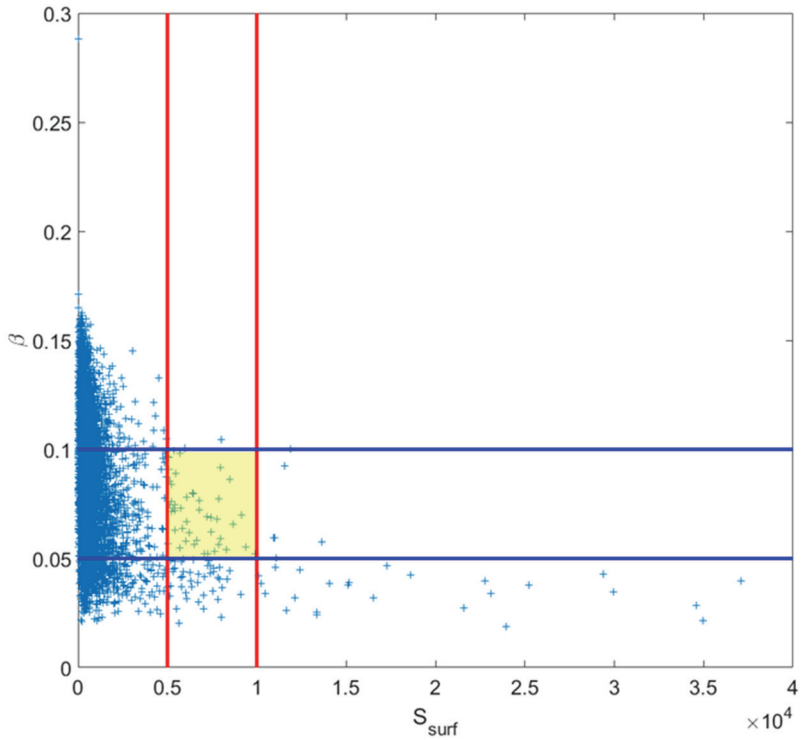
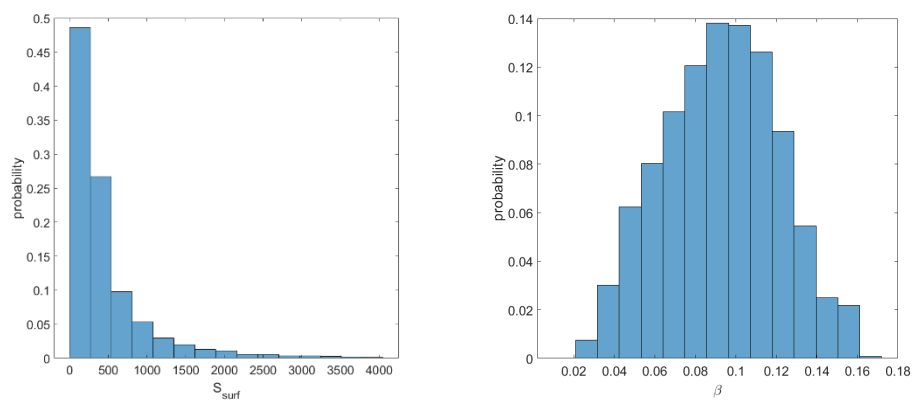


Fig. 33 Particle exporting through the particle shape library in the intersection of surface area and shape index



(a) (b)

Fig. 34 Histogram of particle (a) surface area and (b) shape index of particle shape library

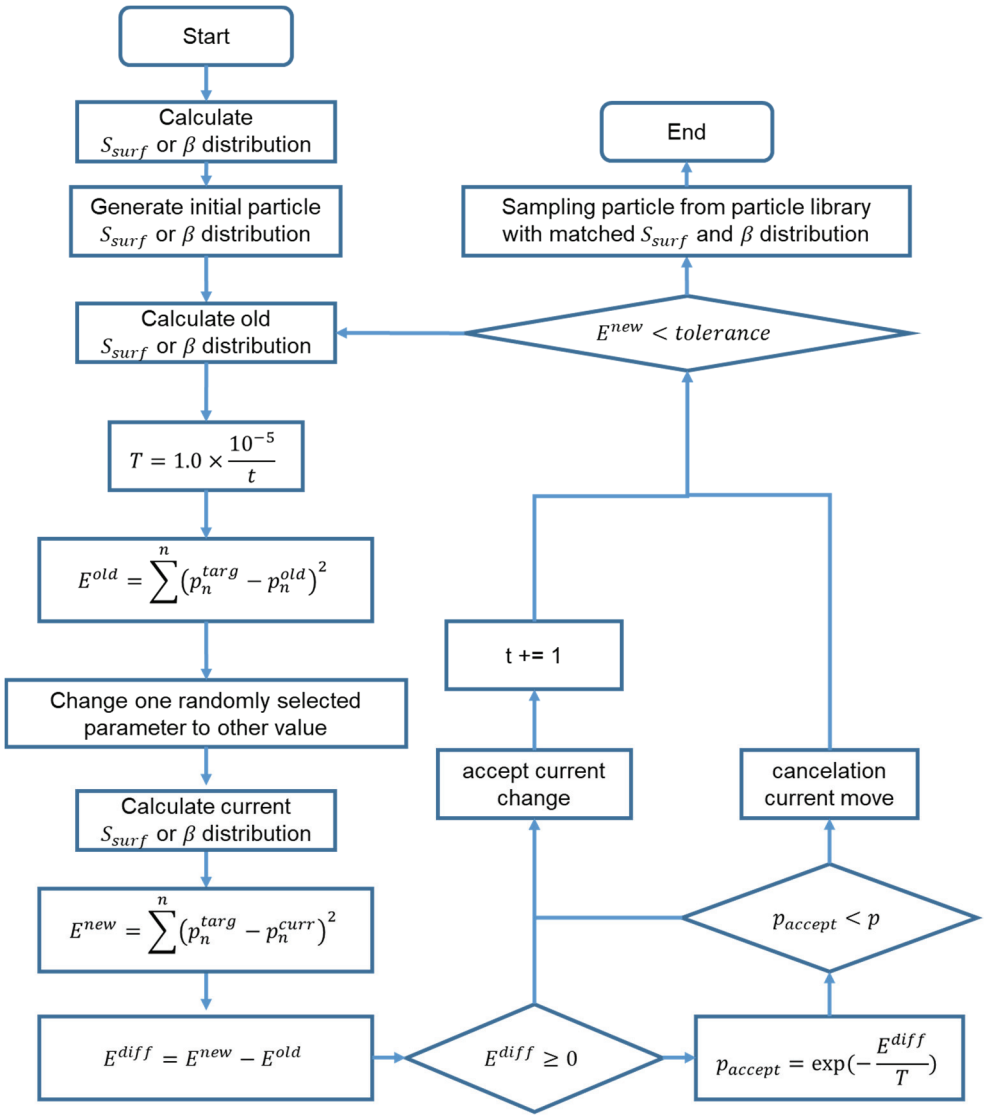


Fig. 35 Flow chart of a particle export algorithm

Algorithm 2. Export particle

```
// function for export particles from the particle shape library
function ShapeMatch (numParticle, target, particleLib)
// numParticle ← number of particles
// target ← contain target data (surface area, volume, principal length)
// particleLib ← particle library, contain particle unit cells
targParameter ← target parameters (surface area and shape index)
reconParameter ← reconstructed parameter (surface area and shape index)
targProb ← target parameter distribution values for each bin
reconProb ← reconstructed parameter distribution values for each bin
error ← least square error of target and reconstruction distribution values
// repeat two times below loop for surface area and shape index
while error>tol && iter < 10000
    rndRemoveParmeter ← remove randomly selected parameter (surface area or shape
                        index)
    rndAddParameter ← add randomly selected parameter (surface area or shape index)
    errorNew ← calculate new error with changed parameter distribution
    errorDiff ← errorNew – error
    if errorDiff >= 0
        probAccept ← exp(-errorDiff/T)
        randProb ← randomly selected value between 0 and 1
        if probAccept <= randProb
            accept the movement of parameter
        end if
            cancel the movement of parameter
    else
        accept the movement of parameter
    end if
end while
exportParticle(particleLib) ← export particles from particle shape library with matched
                              surface area and shape index
```

4.2 Particle Dispersion Matched Map

A particle dispersion is matched through a nearest neighboring distance (NND). The NND is a minimum distance between a particle center with others. The particle center is defined as a particle geometrical centroid. As the definition of NND, each particle has one of NND value, and the two particles at the nearest distance have the same value. The NND of the n-th particle is defined as

$$NND^{\mathrm{n}} = \min\left(\sum_{i=1}^{3}\sqrt{\left(x_i^n - x_i^m\right)^2}\right) \text{ where } \mathrm{n} \neq \mathrm{m} \qquad (20)$$

A particle dispersion is adjusted through matching NND distribution with the target and reconstructed model. The process is similar to the particle sampling algorithm in Section 4.1. A cumulated histogram of target particle NND is generated with ten bins. An initial particle center dispersion is randomly made in the same size as the target image set. One particle center is randomly selected and moved with an arbitrary vector like Fig. 36. The NND for new dispersion is calculated, and the error is calculated with Eq. (17). The SA is applied to optimization particle center position.

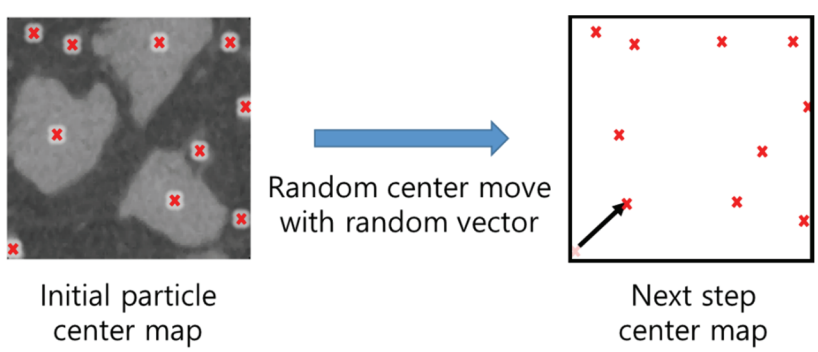


Fig. 36 Schematic diagram for particle center change

In this algorithm, some code lines are added to converge to the result rapidly. When NND is calculated, some particles have out of range that is between a maximum and minimum value of target NND. In that case, these particles are moved until all the NND of reconstructed particles has the same boundaries with the target. From this algorithm, the NND distribution changes from Fig. 37 to Fig. 38 and the *error* moves like Fig. 39. The flow chart of the whole of the dispersion matching algorithm is shown in Fig. 40.

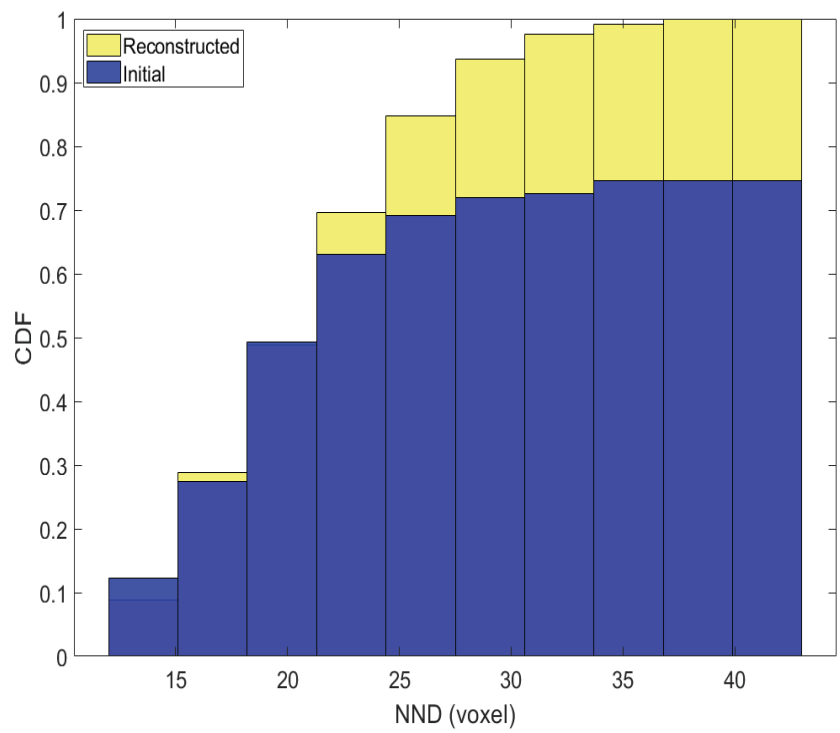


Fig. 37 Histogram of NND distribution of initial model and reconstructed model

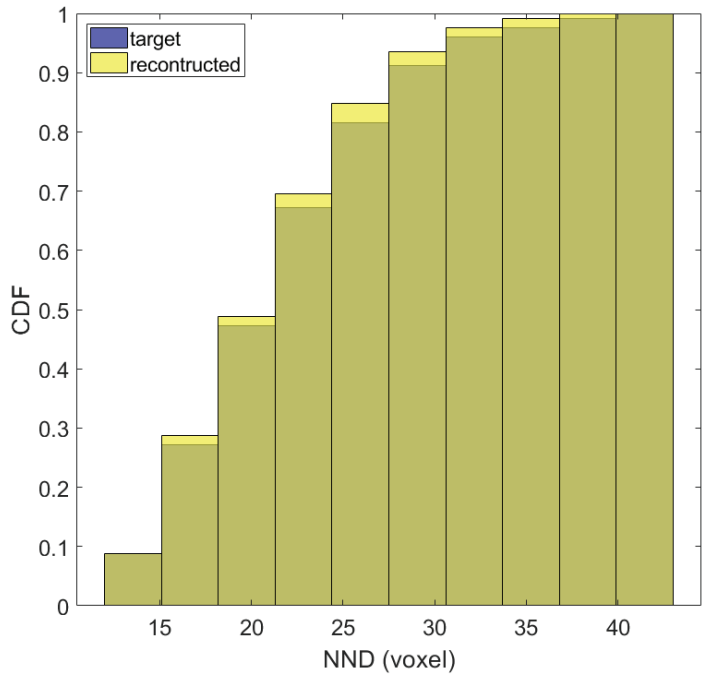


Fig. 38 Histogram of NND distribution of target model and reconstructed model

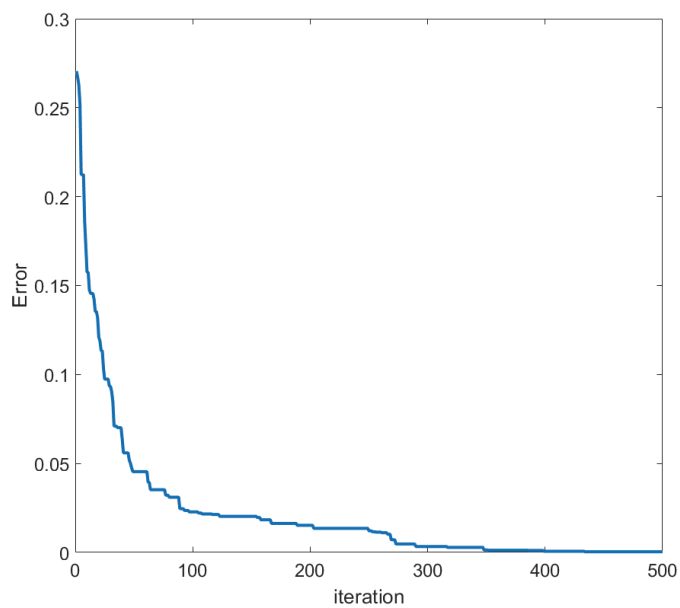


Fig. 39 Error reduction of NND during iteration

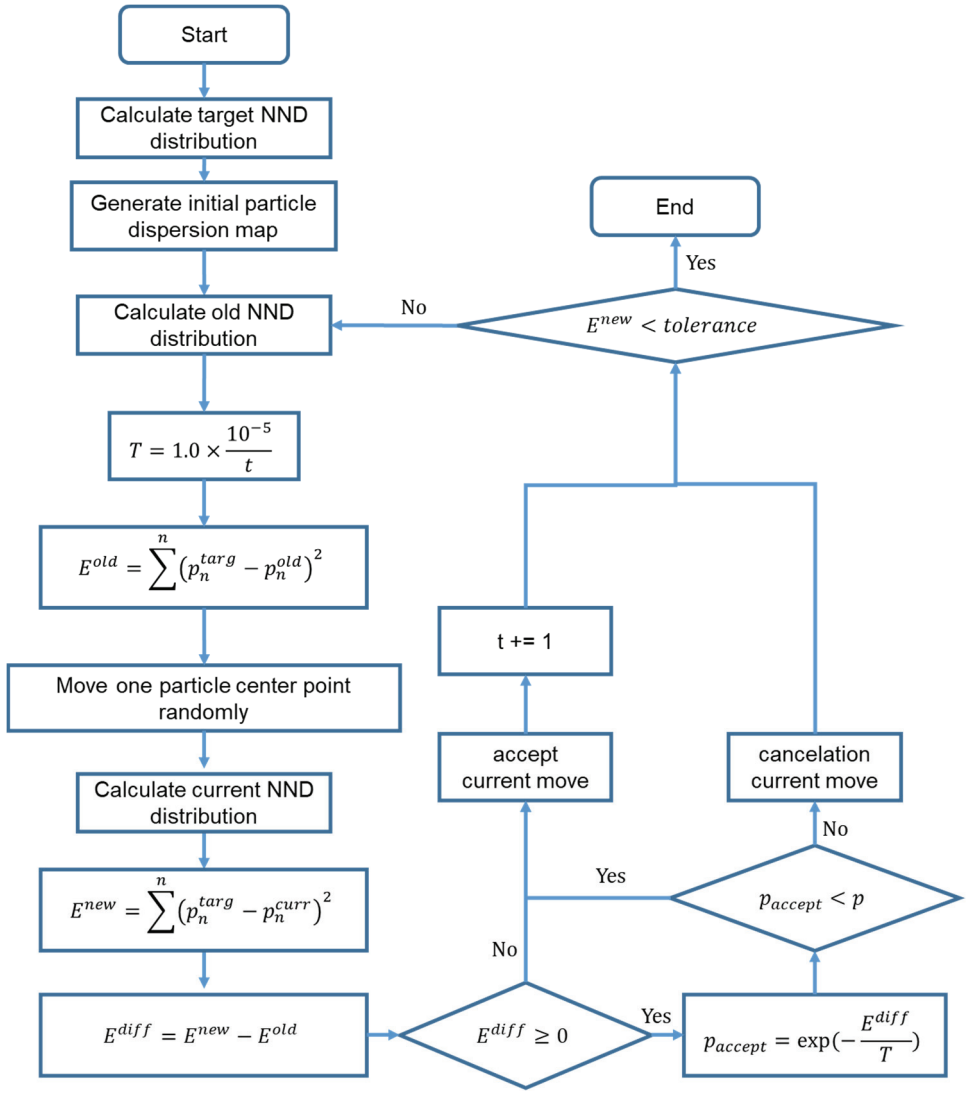


Fig. 40 Flow chart of a particle dispersion adjusting algorithm

Algorithm 3. Particle dispersion matching

```
// function for matching dispersion of particle center
function DisperMatch (imageSize, numParticle, target)
// imageSize ← image size
// numParticle ← number of particles
// target ← contain target data (target center position)
initPosition ← random initial particle position
NND(initPosition) ← calculate initial nearest neighbor distance of initial point
distInitNND ← histogram values of initial NND
error ← least square error of target and reconstruction distribution values
// repeat two times below loop for surface area and shape index
while error>tol && iter < 10000
    rndRemovePoint ← remove randomly selected point
    rndAddNewPoint ← append random point that is not contained in particle center list
     errorNew ← calculate new error with changed parameter distribution
    errorDiff ← errorNew – error
    if errorDiff >= 0
        probAccept ← exp(-errorDiff/T)
        randProb ← randomly selected value between 0 and 1
        if probAccept <= randProb
            accept the movement of parameter
        end if
             cancel the movement of parameter
    else
         accept the movement of parameter
    end if
end while
```

4.3 Particle Packing through Unit Cell Model

The candidates to pack in the reconstructed RVE is exported in Section 4.1, and the particle center map is generated in Section 4.2. In this section, the unit cell that is exported in Section 4.1 is packed follow the particle center map. In this research, particle overlapping is ignored, so in the packing algorithm works for checking to overlap and whether one more particle centers are contained in the particle. When large particles are filled late, many overlapping problems occur. For that reason, it is better to pack large particles first of all, so the particles are sorted as descending order and packed in that order.

A particle center position is randomly selected in the list of particle center map. In this algorithm, a reconstructed RVE is generated in a periodic geometry. For satisfying periodicity, MATLAB built-in function ‘circshift’ that shifts an array circularly is used when particles are packed in the RVE. First, a particle is packed where the vertices of particle unit cell and the RVE coincide. The position vector between the particle center point and the center of a unit cell is calculated, and the ‘circshift’ function move the particle circularly. The process of packing algorithm shown in Fig. 41. When the particles are packed in the RVE, it has to be checked that the two more center points are contained in one particle.

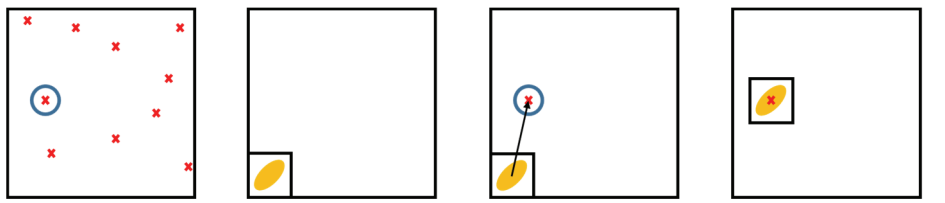


Fig. 41 Schematic diagram of a process to pack particle

The packing algorithm is over when all the exported particles are positioned in the RVE. The reconstructed RVE is checked volume fraction compared with the target volume fraction. If the difference between the two volume fractions is nearly zero, the reconstruction algorithm is ended. However, if the reconstructed volume fraction is not satisfied with target's, first packing algorithm is repeated for ten times, after then the reconstruction algorithm is restarted.

Fig. 42 shows the five target RVEs that are selected in the same specimen and Fig. 43 shows a 2D slice image of the targets. In Fig. 44, five examples of the reconstructed models are presented, one for each of the five target RVEs. Also, Fig. 45 shows a 2D slice image of the reconstructed models.

Algorithm 4. Particle packing

```
// function for packing particle
function PartPacking (imageSize, partCPMap, reconPartVol, reconPartUnitCell,)
// partCPMap ← particle center points coordinate (x, y, and z)
// reconPartVol ← reconstructed particle volume data matrix
// reconPartUnitCell ← reconstructed voxelized particles unit cell dataset that is exported by Algorithm 2
numTotPart = length(reconPartUnitCell) ← total particle number
while partCPMap // end when partCPMap is empty, all particle is packed
    frag = 0 // frag for checking particles overlapping
    while ~frag // end when frag is not 0
        currCP ← randomly selected center point in partCPMap
        unitPartRVE = false (imageSize, imageSize, imageSize) ← define RVE that
                      contains only one particle
        unitPart = reconPartUnitCell(k) ← export one particle from reconPartUnitCell
        unitPartRVE(1:size(unitPart,1), 1:size(unitPart,2),1:size(unitPart,3) = unitPart
        // pack particle on the edge of RVE
        unitPartRVE = circshift(uniPartRVE, currCP - 1 – ceil((size(unitPart)-1)/2))
        // move particle to match current center point with periodicity
        if there are no two more center points in the particle
            if there is no overlapping of particles
                frag = 1
                reconRVE = reconRVE || unitPartRVE // logical 'or' operation
            end if
        end if
    end while
    remove currCP in partCPMap // as a result, length of partCPMap is reduced by one
end while
```

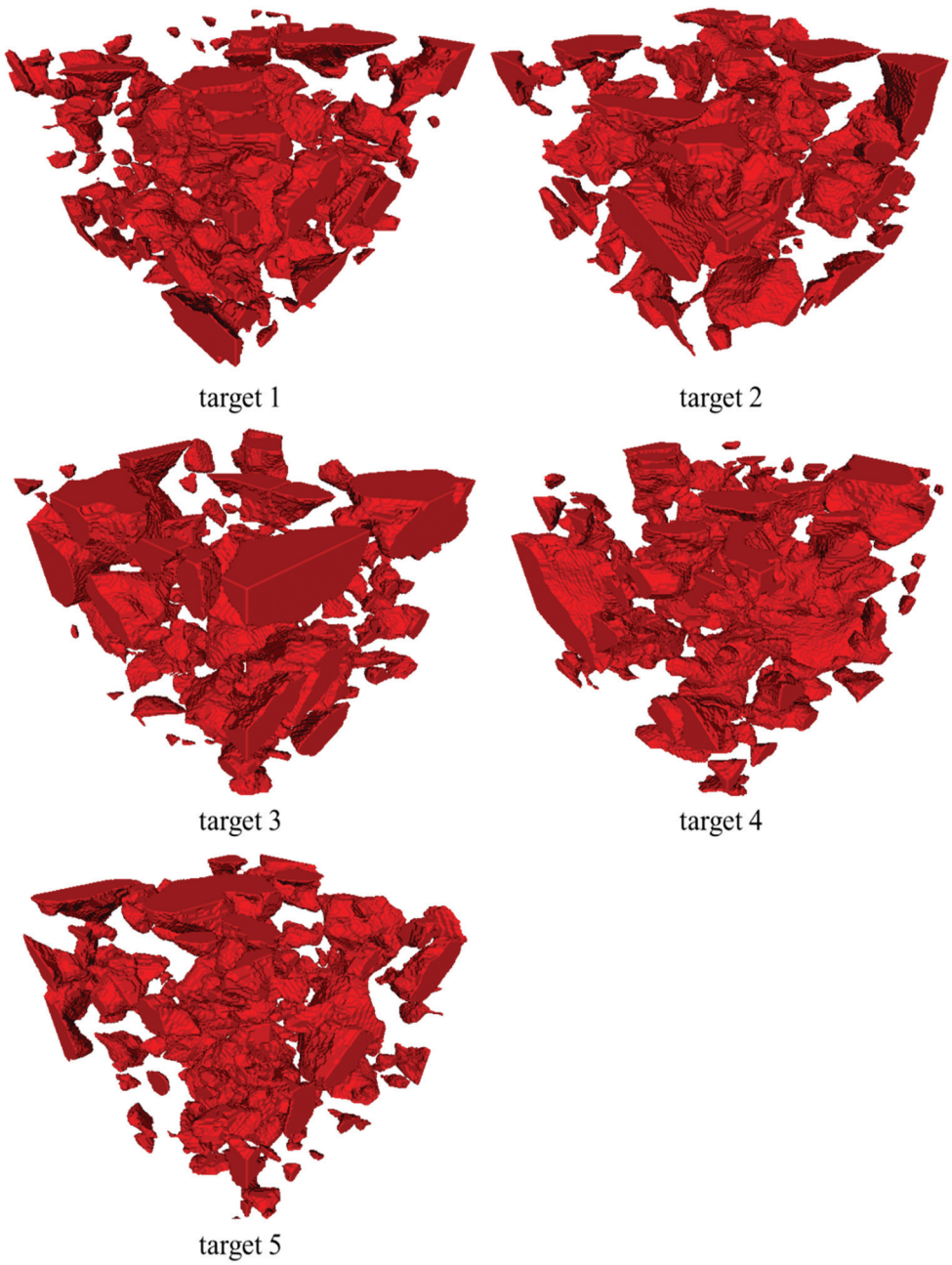


Fig. 42 Target RVE for the five regions of ML specimen

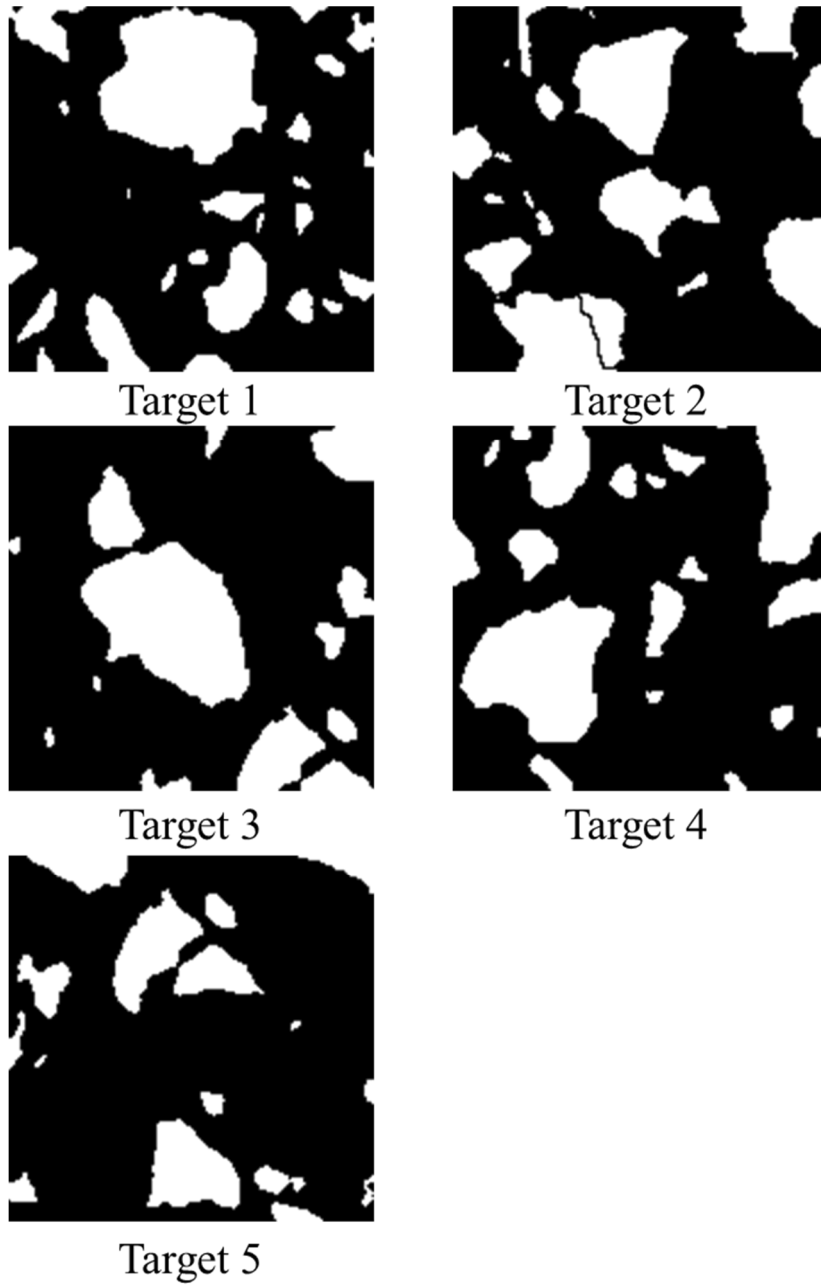


Fig. 43 2D slice image of target RVE for the five regions of ML specimen

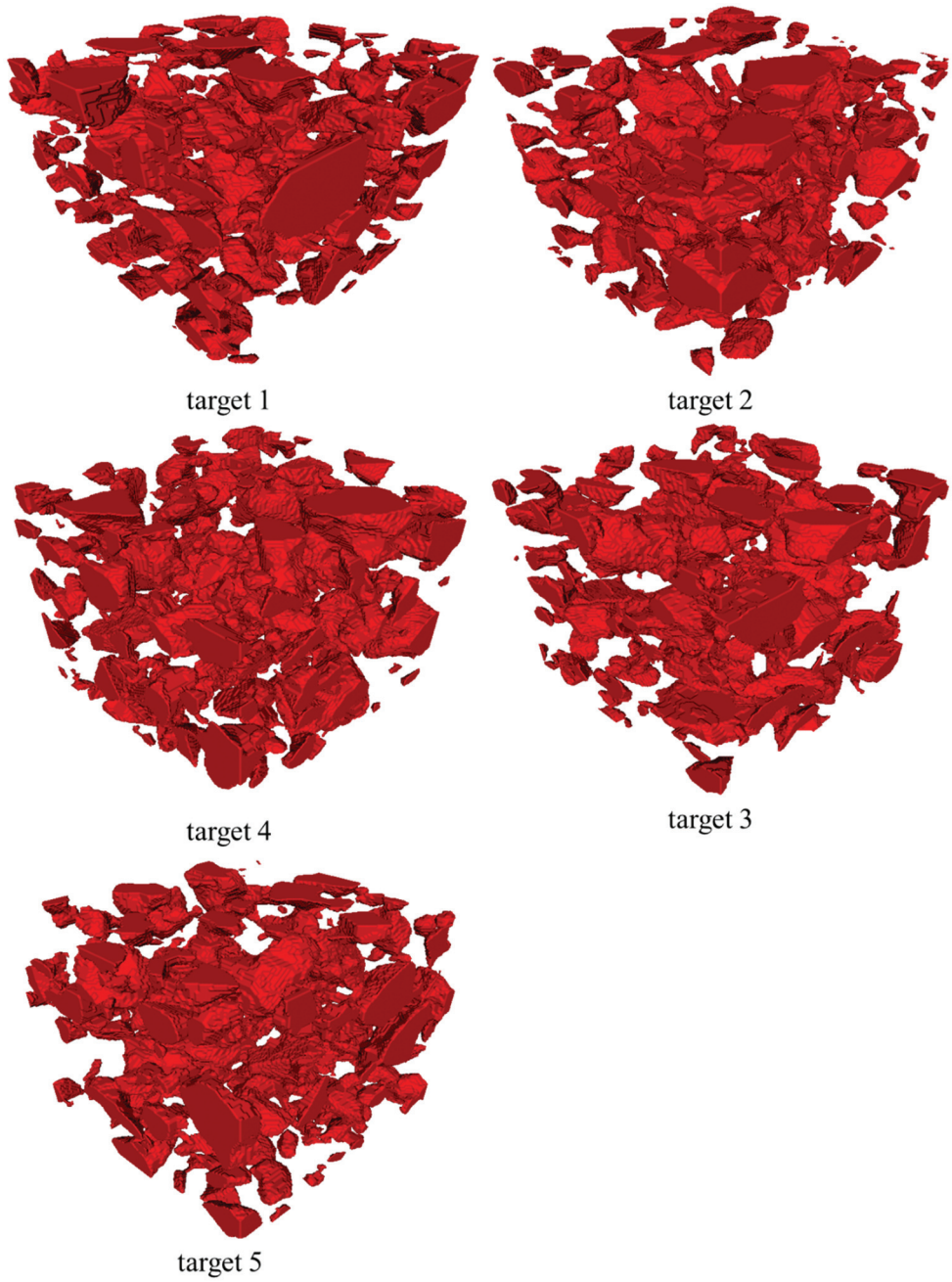


Fig. 44 Reconstructed RVE for the five targets

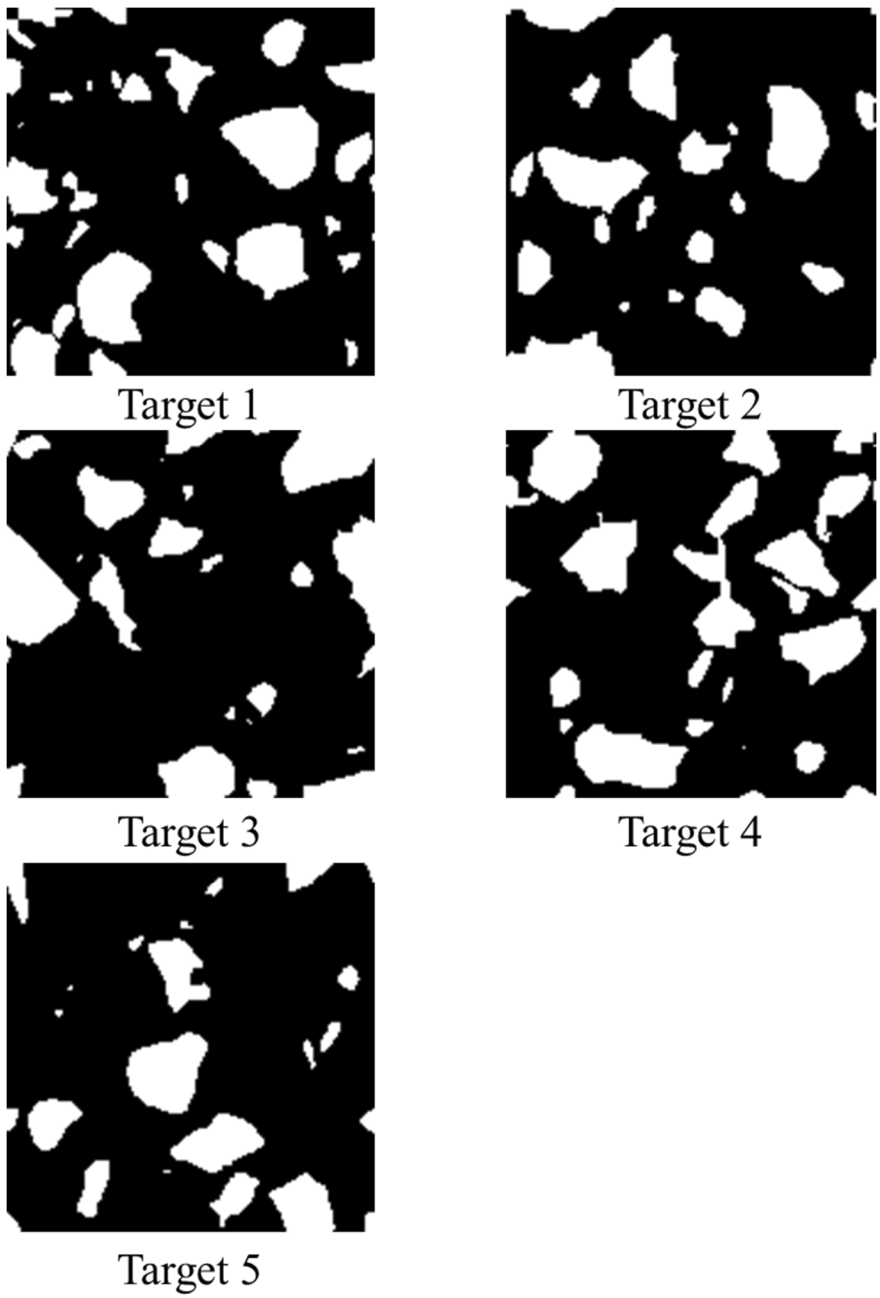


Fig. 45 2D slice image of reconstructed RVE for the five targets

4.4 Verification with Two-Point Correlation Function

A two-point correlation function is calculated with two points in the region of RVE. The function is a function of the distance between two points. In the case of the two-phase composite, there are four components of the two-point correlation function. The synthetic model with two-phases that has a black region, phase 1 (S_1) and a white region, phase 2 (S_2), is shown in Fig. 46. The two-point correlation function is a probability that the two points which have distance *r*, are contained which phases. In this research, the function is calculated with a numerical method and expressed as

$$P_{ij}(\vec{r}) = \frac{N_{ij}}{N}\Big|_{N\to\infty} [\vec{r} = \overrightarrow{r_2} - \overrightarrow{r_1}, (\Phi_1 \in S_i) \cap (\Phi_2 \in S_j)] \qquad (21)$$

where $\overrightarrow{r_1}$ is a position vector of point 1 and Φ_1 is a phase where point 1 is placed. N is the number of total points that are tested in the distance and N_{ij} is the number of point sets that are placed in S_i and S_j. The implementation of TPCF is shown in Fig. 47.

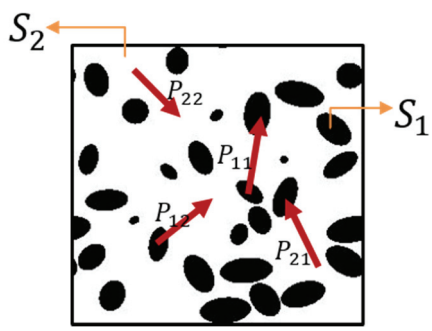


Fig. 46 An example of two-point correlation function in two phases synthetic composites

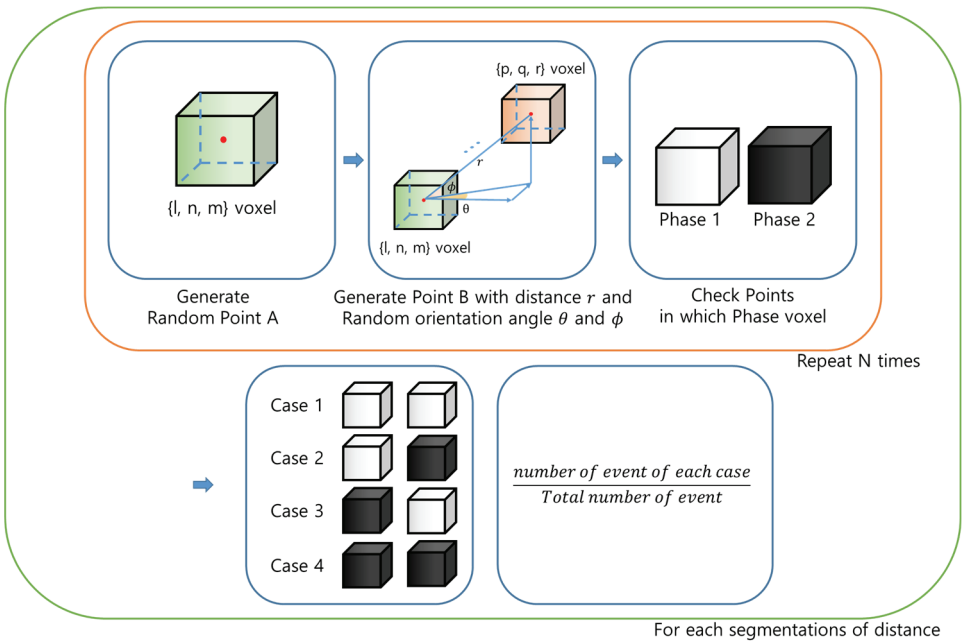


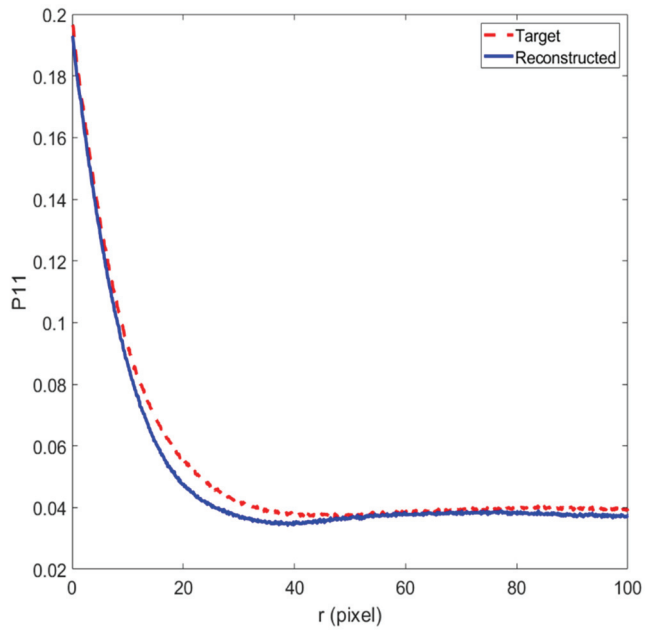
Fig. 47 Implementation for the two-point correlation function

The two-point correlation function has a relation with the phase volume fraction. If distance $\vec{r}$ goes to zero and the two points are in the same phase, the probability is equal to a volume fraction of the phase. Contrastively, if the two points are in a different phase when distance goes to zero, the probability is zero. When distance goes to infinity, the probability is the same as multiple of volume fractions of phase i and j. The relation of distance and volume fraction is expressed as

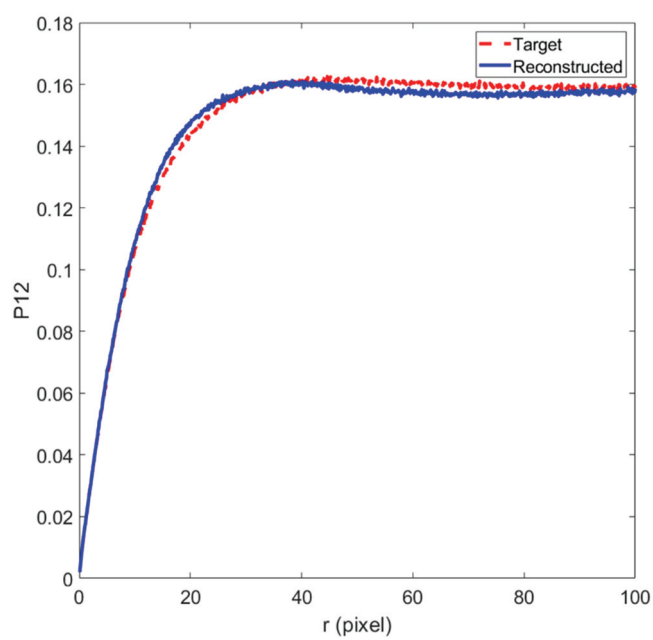
$$\begin{aligned} &\lim_{\vec{r}\to 0} P_{\mathrm{ij}}(\vec{r}) = \nu_{\mathrm{i}}\ (i = j) \\ &\lim_{\vec{r}\to 0} P_{\mathrm{ij}}(\vec{r}) = 0\ (i \neq j) \\ &\lim_{\vec{r}\to \infty} P_{\mathrm{ij}}(\vec{r}) = \nu_{\mathrm{i}}\ (i = j) \end{aligned} \tag{22}$$

where ν_i is volume fraction of i phase.

In this paper, the TPCF is used for verifying the reconstructed microstructure. The image size of the target and reconstructed model is $150 \times 150 \times 150$, however, when a distance between two points goes to 100, the result of TPCF is near to convergence line. For time-consuming and the convergence of result, the TPCF is calculated until the distance is 100. The distance interval is set as 0.1. When the interval decrease, the TPCF result shows correctly, but it needs more time. The number of a set of points for each distance is 500,000. First, one point is located randomly in the region of RVE. Another point is placed with randomly selected Euler angle and a distance. The TPCF is calculated with MATLAB. The result of TPCF is shown in Fig. 48 and Fig. 49.

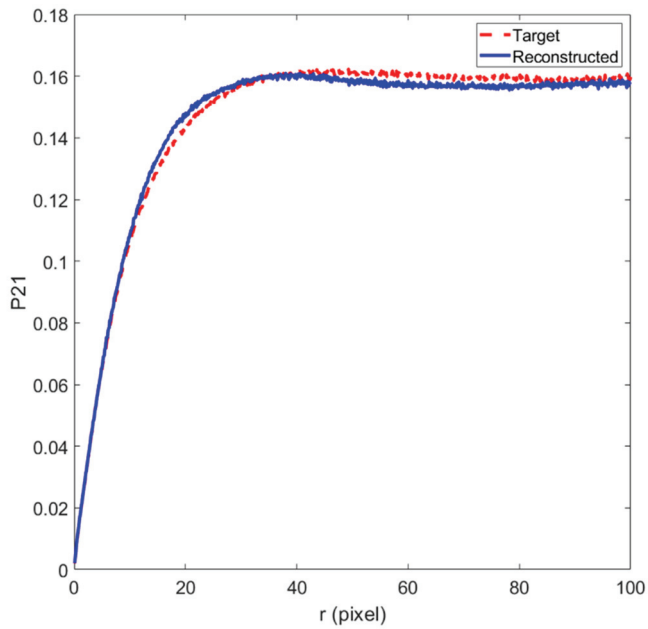


(a)

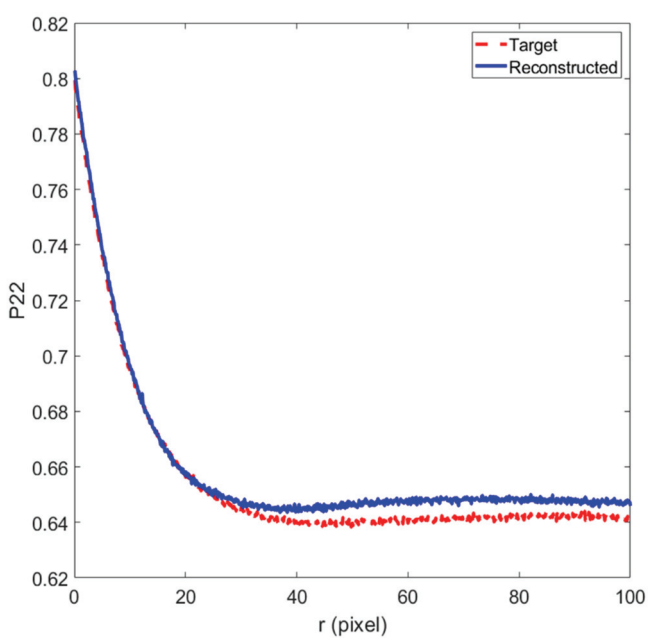


(b)

Fig. 48 TPCF (a) P11 and (b) P12 of target model and reconstructed model



(a)



(b)

Fig. 49 TPCF (a) P21 and (b) P22 of target model and reconstructed model

4.5 New Analytical Two-Points Correlation Function

From the reconstruction model in Section 4.3 and the TPCF calculated with reconstruction model in Section 4.4, a new analytical TPCF equation is developed to find linkage between the morphological parameters and the TPCF using genetic programming (GP). The morphological parameter (the mean and standard deviation of the surface area, shape index, and NND) and the TPCF of each reconstruction are inputted in the GP. The distance between two-point in TPCF is also inputted. In the GP, the distance and morphological parameters are input values, and TPCF is an output value of the analytical equation. The schematic data matrix of one reconstruction model is shown in Fig. 50 with input values in a blue rectangle, and an output value in a red rectangle. The size of the data matrix of one reconstruction model is 1000×8. Moreover, there are fifty reconstruction models in this thesis, so the data matrix size is 50000×8.

r	$E(NND)$	$E(S_{surf})$	$E(\beta)$	$\sigma(NND)$	$\sigma(S_{surf})$	$\sigma(\beta)$	$TPCF(r)$
0.1							
0.2							
⋮							
100							

Fig. 50 The schematic data matrix of one reconstruction model

In this thesis, an open source code in MATLAB, 'GPtips1.0' is used to generate the analytical equation. The program automatically defines the number of generations and trees based on genetic programming. As shown in Fig. 51, the

number of generations is defined ninety-six based on the fitness of the result.

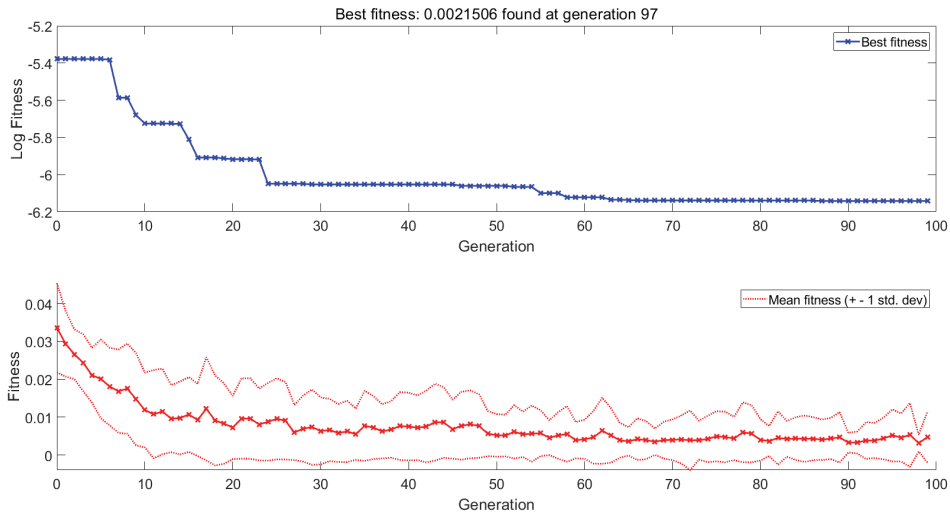


Fig. 51 Changing of fitness with a generation

The program generates an analytical equation based on the training data that is inputted. The new analytical equation of TPCF is following

$$P_{11} = 2.5 \times 10^{-5} r + 2.5 \times 10^{-5} \mu_{NND} + 4.521 \times 10^{-5} \sigma_{NND}$$
$$-0.003754 \sigma_{\beta} - 0.0886 \tanh\left(r \times \tanh\left(\tanh\left(\tanh\left(\mu_{S_{surf}}\right)\right)\right)\right)$$
$$-2.979 \times 10^{-5} \tanh\left(tanh\left(\mu_{S_{surf}} + \sigma_{NND}\right)\right) \quad (23)$$
$$-2.979 \times 10^{-5} \sigma_{NND} \tanh\left(r \mu_{S_{surf}}\right) + 0.0696$$

where r is the distance of two-point in TPCF, $\mu_{S_{surf}}$ is the mean of surface area, μ_{NND} is the mean of NND, σ_{β} is the standard deviation of shape index, and σ_{NND} is the standard deviation of NND. In Fig. 52, the difference between TPCF that is calculated in the numerical method, and the TPCF with the new analytical equation,

Eq. (23).

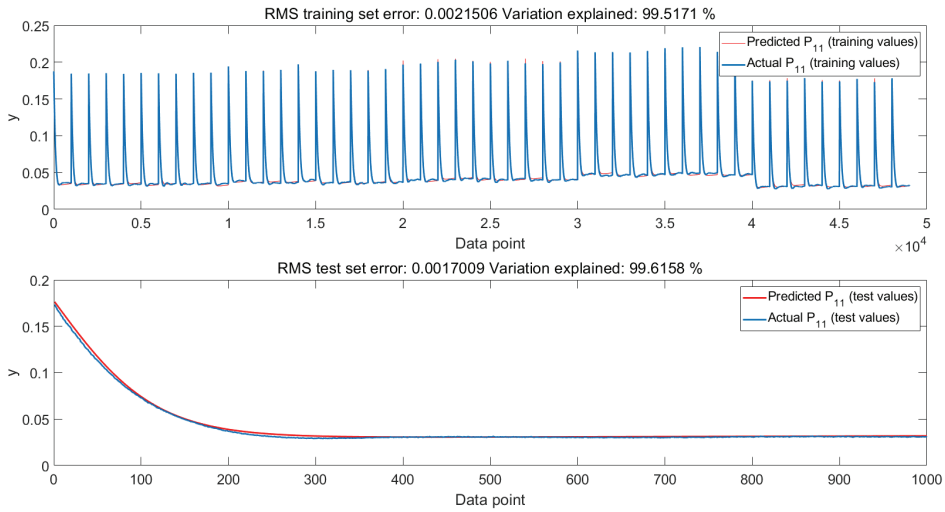


Fig. 52 The difference between numerically calculated TPCF (Actual) and the new analytical equation (Predicted).

In the same method, the new analytical equation of P_{12} $(= P_{21})$ and P_{22} are also generated from the dataset whose output value is P_{12} and P_{22}. The new analytical TPCF of P_{12} is following

$$\begin{aligned} P_{12} &= \tanh\left(2r^2\sigma_{S_{surf}}\tanh\left(\mu_{S_{surf}}\right)\right) - 1.781\times10^{-5}\exp(\sigma_\beta) \\ &-0.3454\tanh\left(\mu_{S_{surf}}\right) + 3.484\times10^{-5}\sigma_{NND}\tanh\left(r\tanh\left(\mu_{S_{surf}}\right)\right) \\ &-\exp\left(-1.855\mu_{S_{surf}}(r+\sigma_{NND})\right)\tanh\left(\mu_{S_{surf}}\right)\times0.6408 \\ &+0.09176 \end{aligned} \tag{24}$$

where $\sigma_{S_{surf}}$ is the standard deviation of the surface area. Also, the equation of P_{22} is following

$$P_{22} = 0.6789 \tanh\left(\mu_{S_{surf}}\right)\left(\sigma_{\beta} - 10.99\right)$$

$$-0.0001051\mu_{NND} \tanh\left(\sigma_{S_{surf}} + r\mu_{S_{surf}}\right) \quad (25)$$

$$+\sigma_{NND}\left(\sigma_{S_{surf}} - \sigma_{\beta}\right) \times 2.017 \times 10^{-5} + 0.07854\mu_{S_{surf}}^{3}\sigma_{NND} + 1.263$$

Chapter 5. Conclusion

This thesis presented a novel investigation on the effect of microstructure morphology on the stress within mechanoluminescent (ML) particles and reconstruction based on morphological parameters. For microstructure characterization, total thirteen morphological variables were defined for this study. X-ray CT images of high-weight fraction real ML composites were used. For segmentation of the irregularly shaped ML particles, a series of efficient digital image processing techniques was applied. The 3D voxel-based morphology analyses extracted all morphological variables with a segmentation algorithm for 3D voxels. A 3D FE model was developed after mesh-sensitivity analysis. The resulting 3D FE model provides VAS. By pairing morphological variables with VAS of 971 individual ML particles, correlation analyses and PCA were conducted. According to statistical analysis results, the surface area, and the shape index turned out to be the statistically independent microstructure-sensitive design variables for ML composites.

The microstructure reconstruction was also carried out using the statistical analysis result: the surface area and the shape index. Moreover, NND is added for matching dispersion of microstructure. For reflecting the real shape of the particle,

particle shape library that is consisted with voxelized unit particle RVE is suggested. The optimization algorithm, SA conducted the reconstruction algorithm. The reconstructed RVEs were verified through TPCF that was implemented as a numerical method.

As already known, morphological descriptor based reconstruction algorithm had an advantage on time cost compared with correlation function based one. This reconstruction algorithm also based on morphological parameters, so it had an advantage on time cost. Also, this algorithm made realistic particle shape using the particle shape library. However, this algorithm had a limitation that is a little difference in TPCF with the target model. It means that there is a little gap between the two models.

Moreover, based on the morphological parameters and TPCF of reconstruction model, the new analytical equation of TPCF is generated with GP. The analytical TPCF equation figure out the relationship between the morphology of components and TPCF.

The result of this research generates a lot of statistically equivalent RVE through characterization of microstructure based on morphological parameters. Many dataset of reconstructed RVE can be used for a novel data-driven science. In the future work, a linkage between microstructure morphological descriptor and microstructure properties will be analyzed based on the reconstruction dataset.

Bibliography

[1] Fu S-Y, Feng X-Q, Lauke B, Mai Y-W. Effects of particle size, particle/matrix interface adhesion and particle loading on mechanical properties of particulate–polymer composites. Composites Part B: Engineering. 2008;39(6):933-61.

[2] Kumar H, Briant CL, Curtin WA. Using microstructure reconstruction to model mechanical behavior in complex microstructures. Mechanics of Materials. 2006;38(8-10):818-32.

[3] Torquato S. Random heterogeneous materials: microstructure and macroscopic properties: Springer Science & Business Media; 2013.

[4] Pant LM, Mitra SK, Secanell M. Stochastic reconstruction using multiple correlation functions with different-phase-neighbor-based pixel selection. Physical Review E. 2014;90(2):023306.

[5] Jiao Y, Stillinger FH, Torquato S. A superior descriptor of random textures and its predictive capacity. 2009;106(42):17634-9.

[6] Torquato S, Lu B. Chord-length distribution function for two-phase random media. Physical Review E. 1993;47(4):2950-3.

[7] Lu B, Torquato S. Lineal-path function for random heterogeneous materials. Physical Review A. 1992;45(2):922-9.

[8] Torquato S, Beasley JD, Chiew YC. Two□point cluster function for continuum percolation. 1988;88(10):6540-7.

[9] Al-Ostaz A, Diwakar A, Alzebdeh KIJJoMS. Statistical model for characterizing random microstructure of inclusion–matrix composites. 2007;42(16):7016-30.

[10] Tewari A, Gokhale AM. Nearest-neighbor distances between particles of finite size in three-dimensional uniform random microstructures. Materials Science and Engineering: A. 2004;385(1):332-41.

[11] Yang S, Tewari A, Gokhale AM. MODELING OF NON-UNIFORM SPATIAL ARRANGEMENT OF FIBERS IN A CERAMIC MATRIX COMPOSITE. Acta Materialia. 1997;45(7):3059-69.

[12] Wang W, Dai Y, Zhang C, Gao X, Zhao M. Micromechanical Modeling of Fiber-Reinforced Composites with Statistically Equivalent Random Fiber Distribution. Materials. 2016;9(8):624.

[13] Bostanabad R, Zhang Y, Li X, Kearney T, Brinson LC, Apley DW, et al. Computational microstructure characterization and reconstruction: Review of the state-of-the-art techniques. Progress in Materials Science. 2018;95:1-41.

[14] Yeong C, Torquato S. Reconstructing random media. II. Three-dimensional media from two-dimensional cuts. Physical Review E. 1998;58(1):224.

[15] Okabe H, Blunt MJ. Pore space reconstruction using multiple-point statistics. Journal of Petroleum Science and Engineering. 2005;46(1):121-37.

[16] Xu HY, Dikin DA, Burkhart C, Chen W. Descriptor-based methodology for statistical characterization and 3D reconstruction of microstructural materials. Computational Materials Science. 2014;85:206-16.

[17] Yang M, Nagarajan A, Liang B, Soghrati S. New algorithms for virtual reconstruction of heterogeneous microstructures. Computer Methods in Applied Mechanics and Engineering. 2018;338:275-98.

[18] Xu H, Liu R, Choudhary A, Chen W. A Machine Learning-Based Design Representation Method for Designing Heterogeneous Microstructures. Journal of Mechanical Design. 2015;137(5):051403--10.

[19] Sundararaghavan V, Zabaras N. Classification and reconstruction of three-dimensional microstructures using support vector machines. Computational Materials Science. 2005;32(2):223-39.

[20] Bostanabad R, Bui AT, Xie W, Apley DW, Chen W. Stochastic microstructure characterization and reconstruction via supervised learning. Acta Materialia. 2016;103:89-102.

[21] Kim Y, Yun GJ. Effects of microstructure morphology on stress in mechanoluminescent particles: Micro CT image-based 3D finite element analyses. Composites Part A: Applied Science and Manufacturing. 2018;114:338-51.

[22] Fujino Y, Xu CN, Nishibori M, Teraoka Y. Development of highly sensitive mechanoluminescent sensor aiming at small strain measurement. JOURNAL OF ADVANCED DIELECTRICS. 2014;4(2):1450016.

[23] Jolliffe IT. Principal Component Analysis. 2nd ed. New York: Springer-Verlag; 2002.

국문초록

형상 라이브러리를 이용한 응력발광 복합재료의 형태학적 특징 기반 미세구조 특성화 및 재구성

김 영 환

기계항공공학부

서울대학교 대학원

이 논문에서는 응력 발광 세라믹 입자강화 복합재료에 대한 새로운 형상 특성에 의한 미세구조 특성화와 재구성이 진행되었다. 응력발광 세라믹 소재(Mechanoluminescent, ML)은 응력세기에 비례해서 빛을 방출한다. 이에 따라 응력발광 소재는 응력 센서로 활용하기 위한 연구가 진행되고 있다. 응력발광 소재가 빛 강도는 편향응력(deviatoric stress)에 비례한다는 점은 이미 알려져 있다. 응력발광 소재를 설계하는데 있어서, 입자가 받는 응력을 높일 수 있는 형상 특성을 찾는 것이 중요하다. 이런 선상에서 이 연구의 목적은 응력에 가장 민감하게 반응하는 형상 변수를 부피 평균한 폰 미제스 응력(volume averaged von Mises stress)과 형상 변수 사이의 통계적 상관관계 분석을 통해서 찾는

것이다. 응력발광 소재의 실제 형상을 확인하기 위해서 X선 마이크로컴퓨터 단층촬영으로 응력 발광 세라믹 복합재료의 단층 촬영해서 이미지를 확보하였다. 단층촬영 이미지 품질을 개선하고, 입자를 분리하기 위해서 가우시안 필터, 중앙값 필터, 워터셰드 알고리즘과 같은 이미지 처리 알고리즘을 사용하였다. 입자에 대한 13가지 형상 변수를 통해서 미세구조를 특성화했으며, 3차원 유한요소 해석을 통해서 각 입자에서 발생하는 부피 평균 폰 미제스 응력을 계산했다. 유한요소 해석 결과와 입자 형상 해석 결과를 연결해서 데이터베이스를 구축하고, 형상과 응력에 대한 상관관계를 조사했다. 이를 위해서 주성분 분석으로 형상변수와 부피평균응력의 상관행렬에서 독립된 성분을 찾았다. 통계적 해석을 통해서 입자 겉넓이와 shape index가 가장 부피평균응력에 민감하다는 것을 확인하였고, 이를 재구성 변수로 결정했다. 입자 분포는 최인접거리(nearest neighbor distance) 분포를 이용하였다. 이 연구에 도입된 아이디어로 실제 구성을 반영하기 위해서 복셀화된 9,687개 입자 단위 셀로 구성한 입자 형상 라이브러리를 사용했다. 목표 모델 변수와 재구성 모델 변수의 확률 분포를 일치하게 만드는 방법으로 성공적으로 재구성을 수행했다. 분포를 같게 만들기 위해서 최적화 기법인 담금질 기법(simulated annealing)을 사용했다.

재구성한 미세구조를 two point correlation function을 통해서 검증하였다. 그리고 재구성 모델의 데이터세트를 이용하여 형상 변수의 함수로 이루어진 새로운 TPCF 해석식을 도출하였다. 논문에서 제시한 알고리즘은 빠른 계산 속도에 장점이 있었으며, 입자 형상 라이브러리를 사용하여 실제 형상과 유사한 재구성 모델을 구성할 수 있다는 장점이 있다. 이 알고리즘을 통해서 생성한 수많은 미세구조 자료와 데이터 과학을 결합하여 응력발광 소재의 미세구조 설계에 활용을 기대할 수 있다.

주요어: 미세구조 특성화, 미세구조 재구성, 형상 라이브러리, 형상 해석, 3차원 유한 요소 해석, 응력발광소재

학 번: 2017-26274