Local Mechanical Fields in Interface Percolated Nanocomposites

by

Brian James Burrows

Bachelor of Science
Clemson University 2010

Submitted in Partial Fulfillment of the Requirements

for the Degree of Master of Science in

Mechanical Engineering

College of Engineering and Computing

University of South Carolina

2014

Accepted by:

Sarah Baxter, Director of Thesis

Edward Gatzke, Reader

Lacy Ford, Vice Provost and Dean of Graduate Studies

Abstract

Polymer nanocomposites can enable the innovative design of multi-functional materials. Metallic fillers in polymer matrices can exhibit improved electronic properties at low volume fractions while maintaining the low density, transparency, and easy processing of polymers. Surprisingly, mechanical properties also show enhancement at these uncharacteristically low volume fractions. Two mechanisms have been suggested as contributing to this enhancement. The first is the formation of a percolated microstructure; the second is the significant influence of the interface region between the matrix and filler. The majority of mathematical models describing this novel mechanical behavior are based on percolation models, which only consider microstructural connectivity. Changes in mechanical properties are likely to be affected by complex microstructures, beyond the simply connected, as well as by micromechanical mechanisms associated with these microstructures. These more complex microstructures and mechanisms may be challenging to identify and describe. In this work the underlying mechanical mechanisms are investigated using a probabilistic and statistical characterization of local strain fields. These continuous fields are more amenable to statistical characterization than the spatial ternary (matrix, particle and interface) fields that describe the microstructure. An apparent percolation threshold is identified based on statistical characterization of the elastic moduli, distributions of local strains and spatial autocorrelation of local strain fields. The statistics of strain fields associated with microstructures producing minimum and maximum moduli are also compared.

Table of Contents

List of Figures

Chapter 1

Introduction

The mechanical properties of polymer nanocomposites do not increase with volume fraction in the same way that traditional particulate reinforced composites do. Traditionally, first order predictions of the mechanical properties of composites involve weighted averages based on the volume fraction of the phases. However, mechanical properties of nanocomposites have been experimentally observed that are significantly larger than those predicted by these mean field approximations at comparable volume fractions. [1, 2, 3]

The underlying mechanism behind this behavior is believed to be linked to interface region formed in composite materials.[4] An interface region forms in all composite materials through interactions at the boundary between a compliant matrix and a stiffer included phase. These interactions may include the quality of bonding between the material phases, chemistry changes at the bonding site, confinement of the matrix, and/or decreased mobility of polymer chains around the included phase. These constraints on the matrix material may alter the materials stress/strain behavior, resulting in an effectively stiffer region of material around the particle phase. This phenomenon appears to be exaggerated in nanocomposites because of the large surface area to volume ratio at the nanoscale. In addition to contributing to the overall effective stiffness of the composite, it has been hypothesized that these interfacial regions contribute to the formation of connected microstructures either by forming reinforcing structures between particles, a pseudo-percolation, or by percolating themselves.[5]

The sudden large jump in material properties behaves in a manner similar to those seen in percolation models used to study electrical conductivity in nanocomposites. Due to this similarity, there has been an attempt to model effective properties by adapting the power laws used to model percolation in electrical conductivity.

$$\sigma = \sigma_0(\vee - \vee_c)^s, \vee > \vee_c \tag{1.1}$$

$\vee_c$ is the percolation threshold, which is the volume fraction where a connected microstructure is likely to form. The filler volume fraction is denoted by $\vee$ and the percolation exponent s; σ_0 is the electrical conductivity of the filler material, and σ is the effective conductivity of the composite. An ideal percolation curve is shown in figure: 1.1.

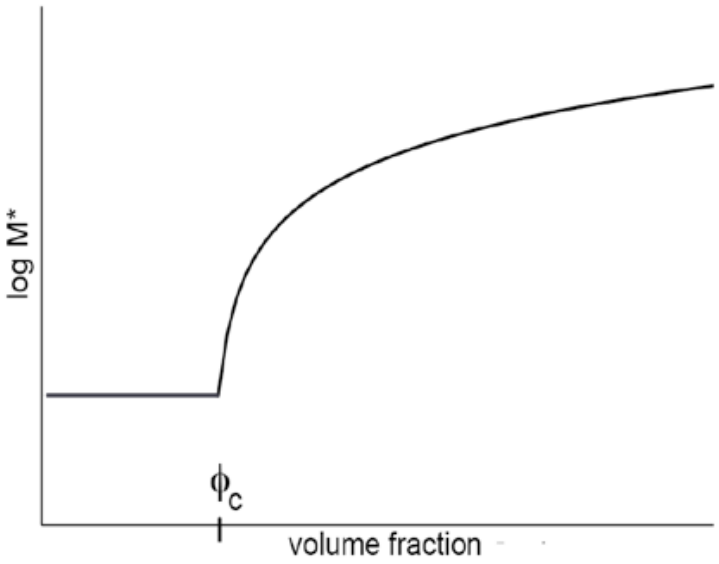

Figure 1.1 An ideal percolation curve

Monte Carlo simulations have been used in order to determine percolation thresholds for various filler geometries and sizes. These models are not mechanics based, but they can be used to predict percolation thresholds based on connectivity [6]. One

model that attempts to predict elastic moduli, E^*, over the entire range of volume fractions is:

$$E^*(\nu) = \begin{cases} E_m(\vee_c - \vee)^{-a} & \text{if } \vee \leq \vee_c \\ E_p(\vee - \vee_c)^f & \text{if } \vee_c < \vee, \end{cases} \tag{1.2}$$

where E_m and E_p are the moduli of the matrix and particle phases and $\vee$ is the volume fraction of the filler.

Other models have been developed for the case of $\vee \geq \vee_c$, such as the phenomenological series/parallel model for composite elastic modulus [7, 8], which takes a more mechanics based approach.

$$E^*(\vee) = \frac{(1 - 2\phi + \phi\vee)E_pE_m + (1 - \vee)\phi{E_m}^2}{(1 - \vee)E_p + (\vee - \phi)E_m} \tag{1.3}$$

In this case ϕ is the volume fraction of the particle phase which is actively reinforcing the matrix, and $\vee$ is still the overall particle volume fraction. This value is defined as:

$$\vee = \nu\left[\frac{(\nu - \nu_c)}{(1 - \nu_c)}\right]^b, \nu_c \leq \nu \leq 1 \tag{1.4}$$

and b is a percolation exponent.

The generalized effective media model [9] also takes a more mechanics based approach.

$$(1 - \vee)\frac{\vee({\sigma_l}^{1/t} - {\sigma_m}^{1/t})}{{\sigma_1}^{1/t} + [(1 - \vee_c)/\vee_c]{\sigma_m}^{1/t}} = 0 \tag{1.5}$$

where $\vee$ is the volume fraction of the particle, and $\vee_c$ is the percolation threshold. In this equation, σ_l and σ_h represent the stiffness of the matrix and the particle phases respectively. σ^* is the effective stiffness of the composite, and t is a percolation exponent.

Both the series parallel and generalized effective media models are based on power law predictions. Each requires prior knowledge of the theoretical percolation thresh-

old. This is problematic in modeling nano-composites because the increase in mechanical properties occurs at lower volume fractions than are predicted using percolation theory. This results in these models being used empirically, and curves are fit to experimental data.

The concetric cyllinder [10] model, and a model developed by Brinson et al [11] take a more mechanics based approach to study the effect of an interface region on material properties. Unlike some approaches, these models surround the particle with an interface region, rather than simply assuming that an interface region is present. However, these models do not predict the sharp rise in the elastic modulus at low volume fractions, possibly due to the lack of a probabilistic component to account for random particle distribution.

A combined approach was developed in [4] which embeded a micromechanics model into a Monte Carlo framework. Random microstructures were analysed using the generalized method of cells (GMC) micromechanics model in order to calculate effective properties. This model has the capability to take into account random particle placement. It also has all microstructural effects embedded into the calculations via the micromechanics model. This allowed the percolation effects to arise naturally through the mechanics associated with the random microstructures without making assumptions about a percolation threshold.

This work expands on the use of this method. The high fidelity generalized method of cells (HFGMC), a 2-D micromechanics model is used to study localized mechanical fields in random polymer nanocomposites. This work examines the use of two statistical analysis methods: probability distribution functions and spatial autocorrelation of strain fields.

In Chapter 2, HFGMC is used to investigate how the effective stiffness of polymer nanocomposites is affected by the addition of a stiff particle phase surrounded by a moderately stiff matrix. This section explains the experimental procedure, selection

of material properties, and RUC size. It also provides a basic overview of HFGMC and GMC. The change of volume fraction of each constituent material with respect to an increase in particle volume fraction is also investigated.

In Chapter 3, a general overview of the traditional definition of theoretical percolation thresholds is presented. The rest of this section presents a method of determining an apparent percolation threshold for mechanical percolation.

In Chapter 4, the principal of maximum entropy is used to analyse local strain fields in each of the composite phases. The theory behind using the principal of maximum entropy for engineering problems is discussed, and the formulation is presented. Probability density functions of the local strain fields within individual constituent materials are used to characterize microstructures that produced the least stiff, and most stiff effective composite moduli. Probability density functions are also compared throughout various volume fraction ranges in the range of the calculated apparent percolation threshold determined in chapter 3.

Chapter 5 explores the use of spatial autocorrelation to analyze local strain fields in selected random microstructures. This section discusses the *xcorr2* function as it is presented in Matlab. It reviews limitations and error associated with the *xcorr2*, and a method for generating autocorrelation functions for doubly-periodic repeating unit cells is presented. Finally, autocorrelation functions for the strain fields are presented at volume fractions in the range of the calculated apparent percolation threshold in maximum and minimum modulus microstructures.

Chapter 6 summarizes the work, and conclusions that were drawn from the results are discussed.

Chapter 2

Model Material System and Micromechanics

2.1 Model Material System

The model nano composite consists of gold nanocubes with an aspect ratio of 1. The metallic nanoparticles are embedded into a polymer matrix. Niklaus and Shea have investigated mechanical and electrical percolation using a similar material system[12].

Each constituent material was assigned isotropic material properties, and was treated as a linear elastic material. The mechanical properties of the gold nanocubes were assigned an elastic modulus of 7.8×10^{10} Pa, a Poisson's ratio of 0.35, and a shear modulus of 2.88×10^{10} Pa. The polymer matrix was assigned an elastic modulus of $7.8\text{x}10^{4}$ Pa, a Poisson's ratio of 0.45, and a shear modulus of $2.69\text{x}10^{4}$ Pa. There are no accurate measures of the mechanical properties or thickness of the interface region of nanocomposites, but it can be assumed that the properties lie somewhere between that of the particle and matrix that surrounds it. Based on this assumption, the interface region was assigned a stiffness equal to the geometric mean of the particle and matrix stiffnesses $\sqrt{E_m E_p}$, and a Poisson's ratio of 0.45.

2.2 Micromechanics

Various micromechanics models have been used to study mechanical percolation in polymer nanocomposites. Mechanical percolation has been previously studied using the generalized method of cells (GMC), with a focus on percolation's effect on effective mechanical properties as well as the presence of an interface region's effect on

percolation-like behavior [4]. This study showed that it was possible to capture the percolation effect on effective mechanical properties by using a model that accounted for probabalistic and mechanics based effects.

For studying the local mechanical effects involved in mechanical percolation, the use of the high fidelity generalized method of cells (HFGMC) was required. The HFGMC model is an extension of the generalized method of cells developed by Jacob Aboudi [14]. Both HFGMC and GMC model the response of composite materials using a periodic repeating unit cell (RUC), which consists of a number of subcells, as shown in figure 2.1 [15]. Each subcell is assigned a constitutive material with distinct material properties. These RUC's are connected using periodic boundary conditions in order to generate a representative volume element (RVE). This allows GMC and HFGMC to model a heterogeneous composite material.

Various micromechanics models such as the generalized self-consistent scheme [16], and the Mori-Tanaka method, [17], are very computationly efficient, and they can predict effective properties of composites accurately. But these models only generate mean fields for the strain in each constituent material. This is not an accurate enough estimation of what occurs locally, because in reality, local fields can vary significantly in the constituents [14]. An important aspect of HFGMC is that it has the ability to calculate more accurate local stress and strain fields at the constituent level while maintaining computational efficiency.

HFGMC main advantage over mean field models is that it can account for the spatial distribution of particles in a randomly dispersed composite. HFGMC also allows for more accurate estimates of local fields by subdividing each microstructure into an arbitrary number of subcells, and strains vary within each subcell. The HFGMC addresses the lack of shear coupling in the GMC, and it also allows elastic stress and strain fields to vary linearly within the HFGMC subcells. This is done using a second-order displacement field [14]. The HFGMC loses some of the GMC's

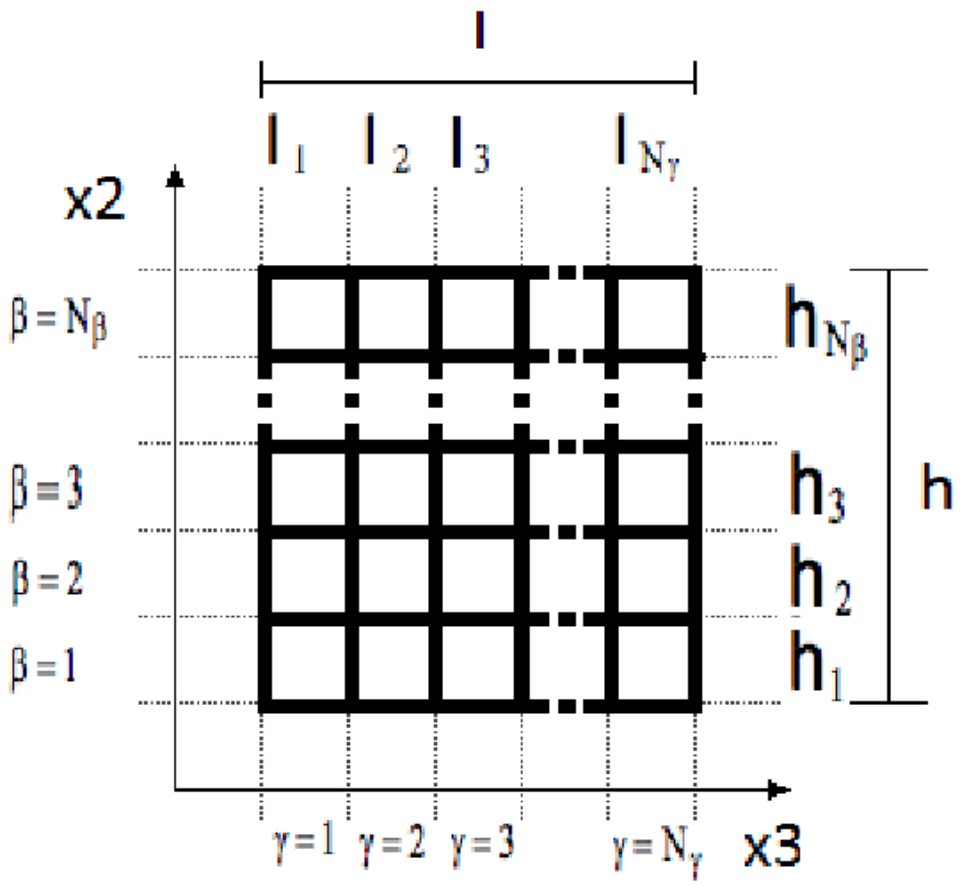

Figure 2.1 Doubly periodic GMC and HFGMC repeating unit cell. The RUC is composed of subcells, each containing a constutuent material. The subcells are identified by the indices β, γ.

computational efficiency by doing this, but it gains accuracy. The HFGMC has been compared to finite element analysis in a wide variety of cases [14], and is able to provide comparable localized fields.

Simulation and Results

Random microstructures were generated in Matlab (See Appendix A) with volume fractions ranging between zero to one hundred percent. Each particle was placed randomly within the microstructure using a Monte Carlo method. Three hundred microstructures were generated at every volume fraction in steps of 5%. The RUC

was a uniformly divided, 48×48, 2-D array, with each particle occupying four subcell spaces (2×2 square).

Each particle was surrounded by an interface region, which was assumed to be half as thick as one particle width. Particles were placed in the RUC in such a way that only matrix and interface subcells could be replaced by a particle subcell. This allows for a consistant particle volume fraction, but created fluctuations in the matrix and interface volume fraction. HFGMC was then used to analyse each microstructure for its effective material properties.

A total of 5,700 microstructures were analysed using HFGMC. Based on the simulation, the effect of volume fraction on the elastic modulus is illustrated in figure 2.2, where the minimum, average, and maximum elastic modulus are plotted versus volume fraction. At ten percent volume fraction, large deviations between the lowest and highest calculated effective elastic modulus are observed. As the particle volume fraction is increased from ten to twenty-five percent, this distribution in properties grows. Above twenty-five percent volume fraction, a narrowing occurs.

From thirty-five to fifty percent particle volume fraction, there is not much change in the difference between minimum and maximum values. Above fifty percent, there is another divergence of properties, which then begins to converge above seventy-five percent. Figure 2.3 shows the changes in the three constituent volume fractions as the particle volume fraction increases. Compared to figure 2.2, a few trends emerge.

In figure 2.3, the volume fractions of the interface and matrix increase and decrease rapidly due to the addition of a few particles. This continues from zero to twenty-five percent volume fraction. This rapid change corresponds to the emergence of large distributions in effective moduli seen in figure 2.2. As the rate of change for matrix and interface volume fractions slows, the spread of mechanical properties begins to taper off. Between thirty to fifty percent particle volume fraction, the interface volume fraction approaches a maximum. This occurs simultaniously to the relatively constant

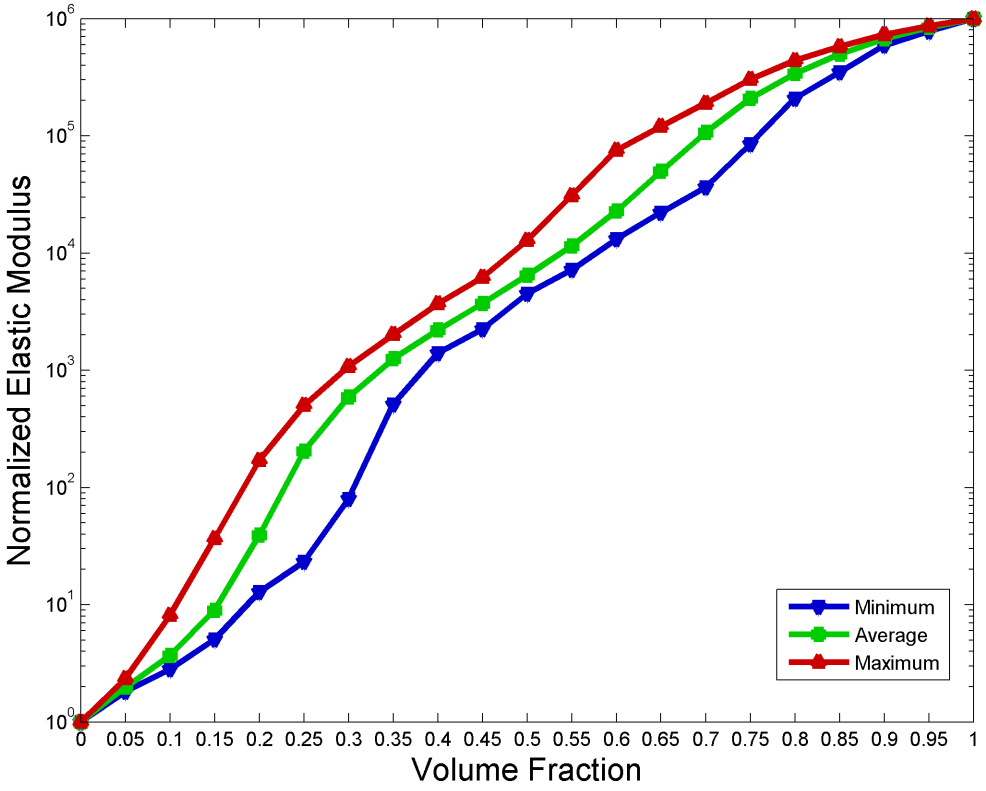

Figure 2.2 The minimum, mean, and maximum effective stiffness plotted versus the particle volume fraction. 300 simulations were analysed at each volume fraction.

distrubution of effective moduli seen in figure 2.2. Around fifty-five to sixty percent particle volume fraction, the composite becomes roughly a two-phase composite on average. This occurs simultaniously with another divergence in mechanical properties in figure 2.2.

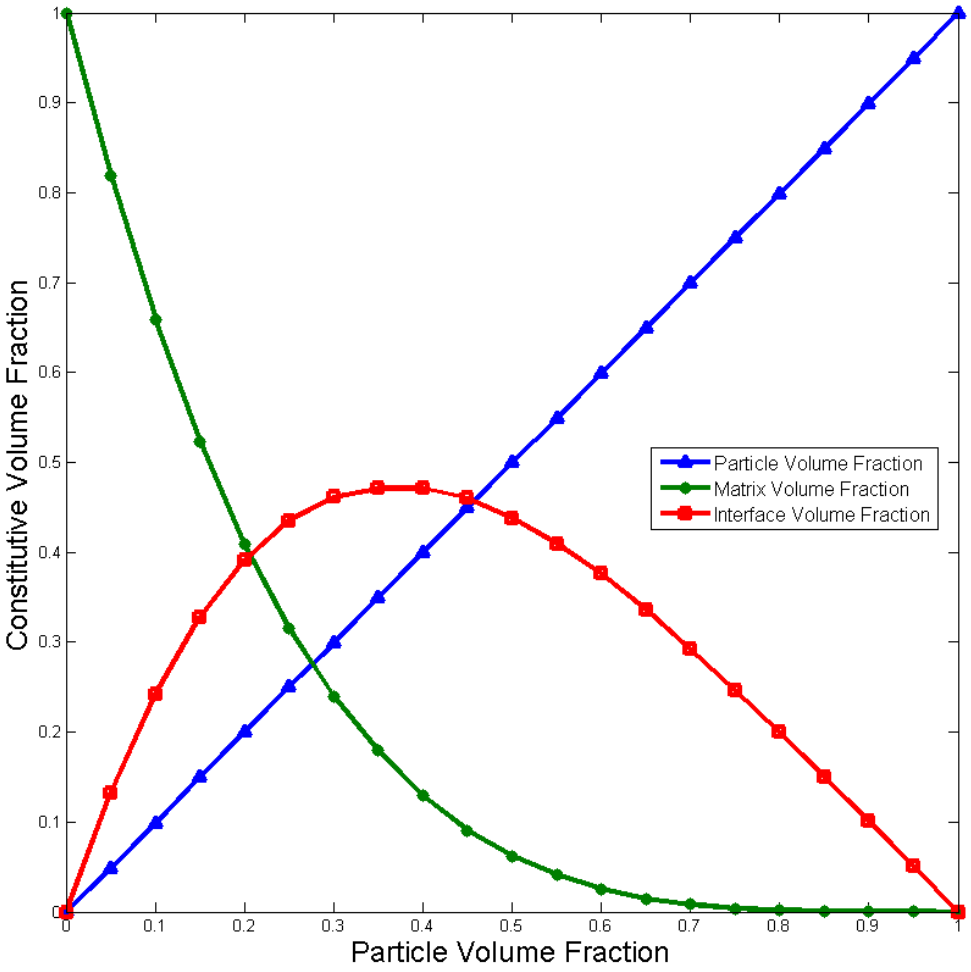

Figure 2.3 The mean volume fractions of each constituent material are plotted versus particle volume fractions.

Chapter 3

Apparent Percolation Threshold

Traditionally, percolation thresholds are defined as the volume fraction where 50 % of simulated microstructures have a connected phase that spans the microstructure. This threshold value is then used in models to predict phenomenon that are dependent on the formation of a connected microstructure. Percolation thresholds have been used to model electrical conductivity in polymer nanocomposites, where the formation of a connected phase allows current to flow through the material.

Generally, material properties which depend on some level of connectedness behave in a binary manner. Below the percolation threshold, the property is not exhibited at all. Once a critical volume fraction is reached, the material begins to exhibit some enhanced property. This is not the case when looking at the relationship between volume fraction and the effective elastic modulus of composite materials. The effective modulus does not depend solely on the formation of a connected microstructure. The addition of any stiffer material to a compliant matrix will result in an overall increase in effective stiffness.

The challenge to modelling nanocomposites is that a significant change in mechanical properties has been observed at very low volume fractions. This occurs well below the volume fraction where classic percolation is predicted. Because the traditional theoretical percolation threshold does not coincide with the volume fraction where there is a sharp increase in mechanical properties, the simulated percolation thresholds are not useful for predictive models. Therefore, the power law models used to capture this rise in mechanical properties must be curve fit to experimental

data where this volume fraction is known, and they contribute little to understanding the physical mechanisms causing this material behavior. However, the concept of pinpointing where a percolation-like effect is likely to occur is still useful.

Towards that goal, an apparent percolation threshold was defined by analysing the statistical skew of the elastic moduli at each volume fraction. Figure 3.1 shows a plot of the average effective elastic modulus plotted versus volume fraction. Overlayed is a graph of the statistical skew of elastic moduli at the same volume fraction.

The skewness, γ_1 of a random variable X is the third standardized moment, and it is defined as:

$$\frac{n}{(n-1)(n-2)} \sum \left(\frac{x_j - \bar{x}}{s} \right)^3 \tag{3.1}$$

where n is the total number of data points, x_j is an element of the data set, $\bar{x}$ is the sample's mean, and s is the standard deviation of the data set.

At very low volume fractions, very few microstructures have significantly enhanced properties due to any microstructural reinforcement. The most significant effect is likely the addition of a stiffer interface. However, with the addition of interface subcells, microstructures that provide mechanical reinforcement may develope. The presence of a small number stiffer microstructures will skew the data set towards a higher mean stiffness. Figure 3.1 shows the skew of effective moduli plotted versus the particle volume fraction. Superimposed is the average effective moduli that was shown in figure 2.2. Figure 3.1 suggests that as volume fraction increases, there are differences due to microstructural effects that emerge at fifteen percent volume fraction. Some microstructures provide better reinforcment than others, causing an increase in mean stiffness that results in a right-skewed data set. As the volume fraction continues to increase from 15-30%, the number of microstructures exhibiting this reinforcing effect also grows. Eventually, it becomes equally likely that a microstructure either exhibits or does not exhibit this reinforcing effect, again resulting in a symmetric, zero skew distribution of effective modulii.

Here, it is assumed that as the microstructural effect on mechanical propeties becomes a factor, the data will display skew as isolated microstructures provide better reinforcement. Eventually, as the number of highly reinforced microstructures increases, the data becomes more symmetric. An apparent percolation threshold is then defined as the volume fraction where the skew of the effective moduli returns to zero.

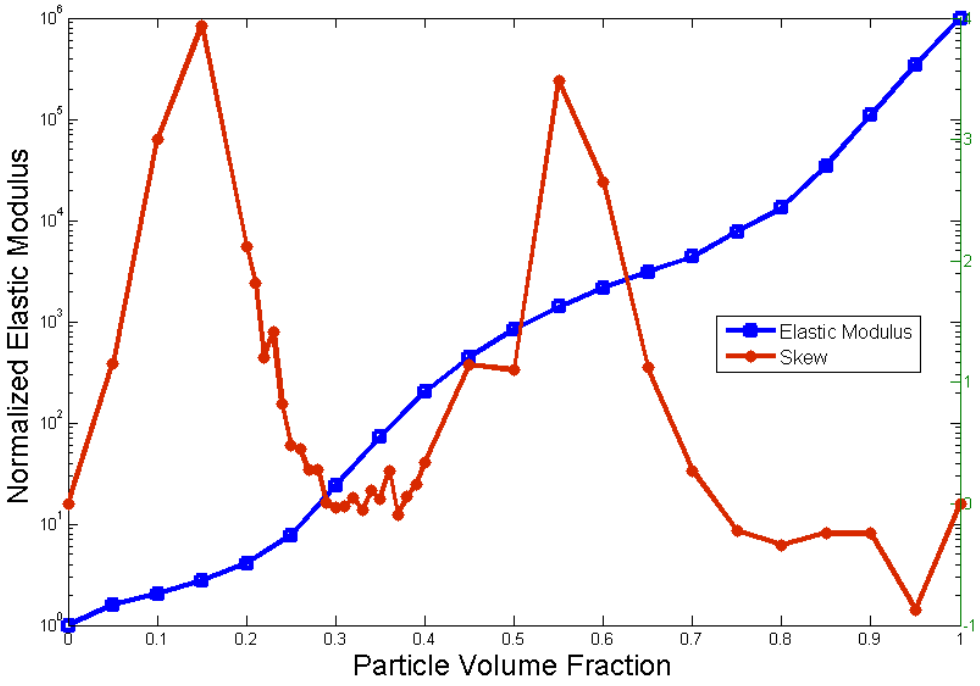

Figure 3.1 A plot of the effective properties and the skew of the data set versus particle volume fraction. The scale on the right represents the skew

When the majority of the microstructures exhibit this percolation effect, a symmetric distribution would be expected to occur. The skew in fig. 3.1 suggests a few interesting things. Between 0 and 15 %, the skew rises rapidly, indicating that there are a few microstructures that are exhibiting significant increases in stiffness. The overall number of reinforced microstructures is small relative to the total, but the stiffness is almost an order of magnitude higher. A highly right-skewed data set indicates the emergence of the mechanism causing above average reinforcement within the

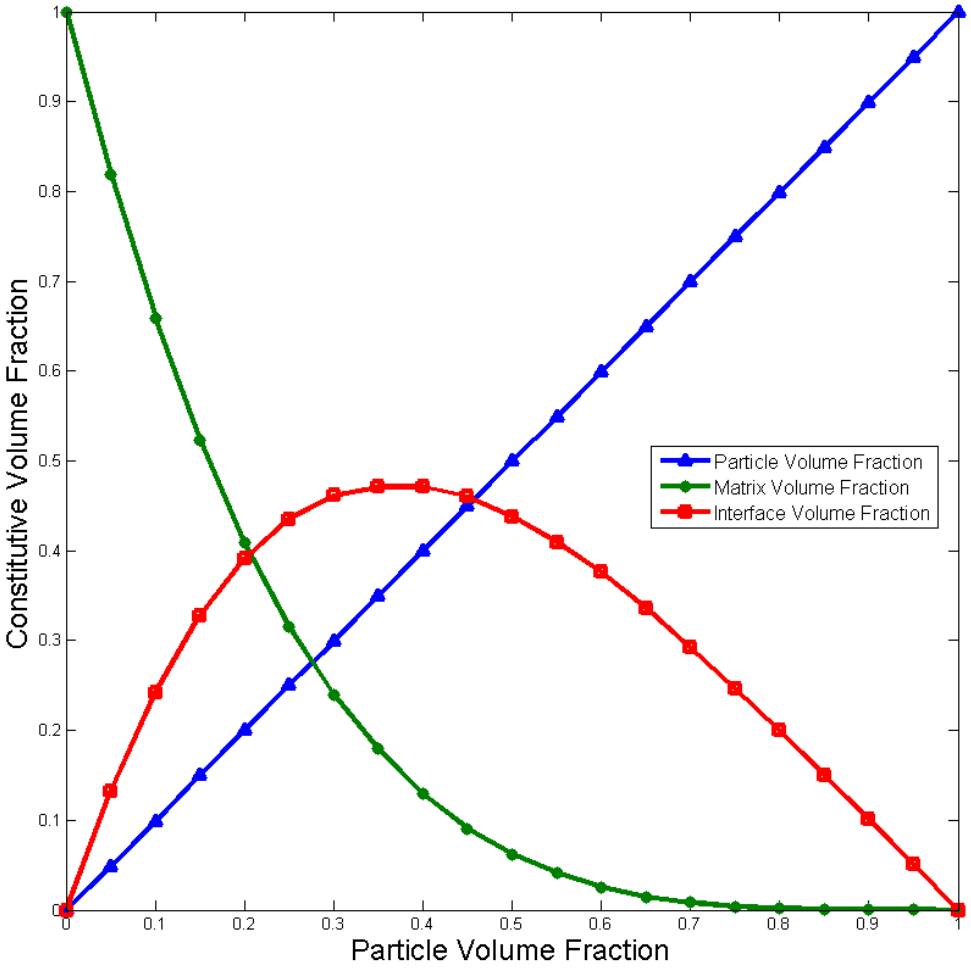

Figure 3.2 The mean volume fractions of each constituent material are plotted versus particle volume fractions.

composite microstructures. When compared to fig. 3.2, this volume fraction range corresponds to the volume fraction where the particle volume fraction is roughly equal to the matrix volume fraction, and the interface volume fraction reaches a maximum.

From 15 to 25 %, the skew begins to decrease. This indicates that significant reinforcement is occuring at a higher frequency. As the particle volume fraction increases, the number of microstructures exhibiting this behavior continues to grow. Above 25 %, the skew begins to taper off gradually as it approaches a zero skew

distribution. The elastic moduli first reach zero skew around 28 % volume fraction. This is when figure 3.2 begins to show that the interface volume fraction is leveling off. Both the skew and interface volume fraction remain relatively constant between 30 and 40 % volume fraction.

Chapter 4

Probability Density Functions of Strain Fields

In addition to the effect of adding a third phase, the interface, into the mechanics of the composite, it seems clear that the reinforcing effect of the phases contributes to the stiffness. It also seems clear that the structure and arrangement of the phases affects the reinforcement of the composite. However, it is difficult to associate variations in elastic moduli with specific microstructural structures. The structures which might be linked to these effects are difficult to visualize, and they are likely more complicated than simple connectivity. So rather than looking at microstructures for patterns and structures, this work proposes tracking the reinforcing effects of the microstructures by statistically analyzing the resulting strain fields.

Probability density functions of the strain fields were constructed using the Principle of Maximum Entropy (PME) [18]. Volume fractions around the apparent percolation threshold were chosen, and microstructures from these volume fractions were selected that produced lowest and highest effective elastic modulus, respectively. Each microstructure was subjected to a load of 1% strain, and the strain field data was separated into strains from each constituent material.

4.1 Principle of Maximum Entropy

PME states that of all possible solutions, one should choose the one that maximizes an entropy term and satisfies all *a priori* conditions. This results in a solution with the maximum uncertainty, or the minimum embedded bias.

The principle of maximum entropy uses the information entropy functional, which

is given by: [19]

$$H[p(\epsilon|J)] = -\int_{-\infty}^{\infty} p(\epsilon|J)\ln(p(\epsilon|J))d\epsilon. \tag{4.1}$$

Where $H[p(\epsilon|J)]$ is the entropy of the data set. The term $p(\epsilon|J)$ is the probability that the local strain, ϵ, will take a certain value at a specific material state, J. This material state takes into account anything that could effect the local strains within the material. This includes constituent volume fractions, constituent geometry, constituent locations, etc.

The moduli are assumed to be bounded, with a minimum

$$\epsilon^- \leq \epsilon \leq \epsilon^+ \tag{4.2}$$

and the mean value, $\bar{\epsilon}$ is assumed to be known. This information is enough to define a first-order PDF, which satisfies the conditions,

$$\int_{-\infty}^{\infty} p(\epsilon|J)d\epsilon = 1 \tag{4.3}$$

0^{th} moment is normalization

$$\int_{-\infty}^{\infty} p(\epsilon|J)\epsilon d\epsilon = \bar{\epsilon} \tag{4.4}$$

1^{st} moment is the mean, $\bar{\epsilon}$

Higher order PDF's must satisfy conditions based on higher order moment equations. This problem can be solved using Lagrange multipliers, and the general form of the N^th order probability distribution function is:

$$P(\epsilon|J) = \begin{cases} P(\epsilon|J) = \exp(\lambda_0 + \sum_{n=1}^{N} \lambda_n \epsilon^n) \text{ for } \epsilon^- \leq \epsilon \leq \epsilon^+ \\ p(\epsilon|J) = 0, \text{ otherwise.}, \end{cases} \tag{4.5}$$

4.2 Results

This work focused on volume fractions near the theoretical, apparent percolation threshold, $\vee_c$, that was determined in the previous chapter. The strain fields for various microstructures were obtained using HFGMC. Probability distribution functions (PDF) of the strain fields in the interface and matrix phases for sample microstructures between volume fractions of 0.22 and 0.34 are shown in fig. 4.1-4.2. Strain values in the particle subcells were distributed over a very small range of values. The PDF's for the particle strains were almost single values, so they have been omitted from the plots.

Figure 4.1 shows the PDFs for the matrix and interface strains pulled from the microstructures that resulted in the maximum effective moduli at particle volume fractions 0.22, 0.25, 0.28, and 0.31. The strains in the interface region are distributed over a much smaller range than the strains in the matrix phase in general. This is consistent with the constraints expected to exist in the interface region due to its attachment to a stiff particle phase. The average strain in the interface region increases from below the global applied strain until it is almost equal to it. The PDFs of the matrix phases show less consistent trends as the volume fraction is increased. The widening distribution suggests a reinforcing effect by stiffer phases, and the shift to higher strains may be the result of the lower matrix volume fraction.

Figure 4.2 shows a comparison of the PDFs in the interface and matrix phases for microstructures that produced the minimum and maximum effective moduli at particle volume fractions of 0.22, 0.25, 0.28, and 0.31. The interface strains for the maximum microstructure are distributed over a larger range than in the minimum microstructures, although at volume fractions above 0.28 the ranges are similar. The mean strain value in the interface region is higher in the maximum microstructures than the minimum microstructures. Mean values in the matrix material did not vary much between the minimum and maximum microstructures.

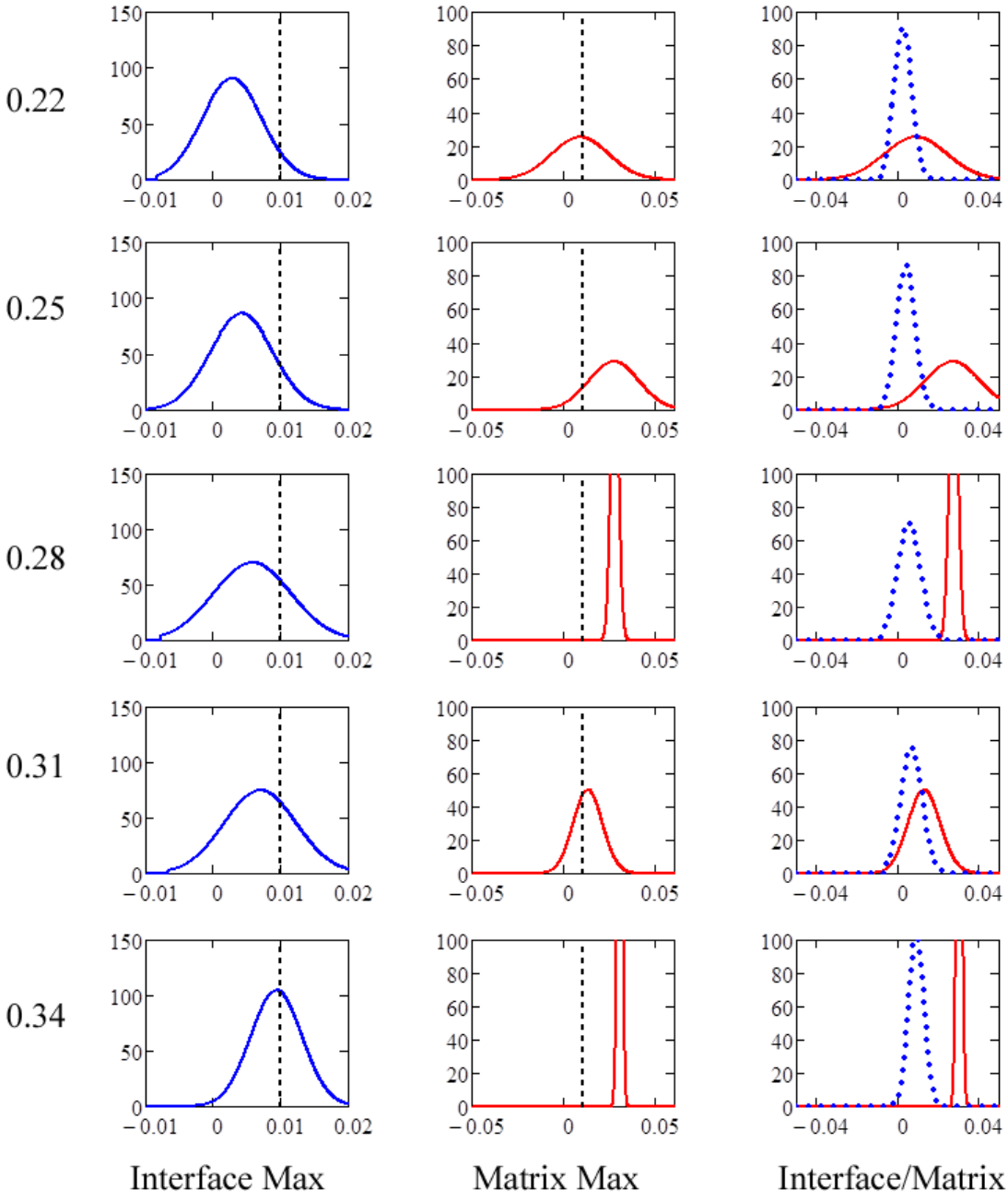

Figure 4.1 Probability distribution functions of the strains in each constitutive material for various maximum modulus microstructures around the percolation threshold volume fraction. 0.22, 0.25, 0.28, 0.31, and 0.34 represent the particle volume fraction of the microstructure studied.

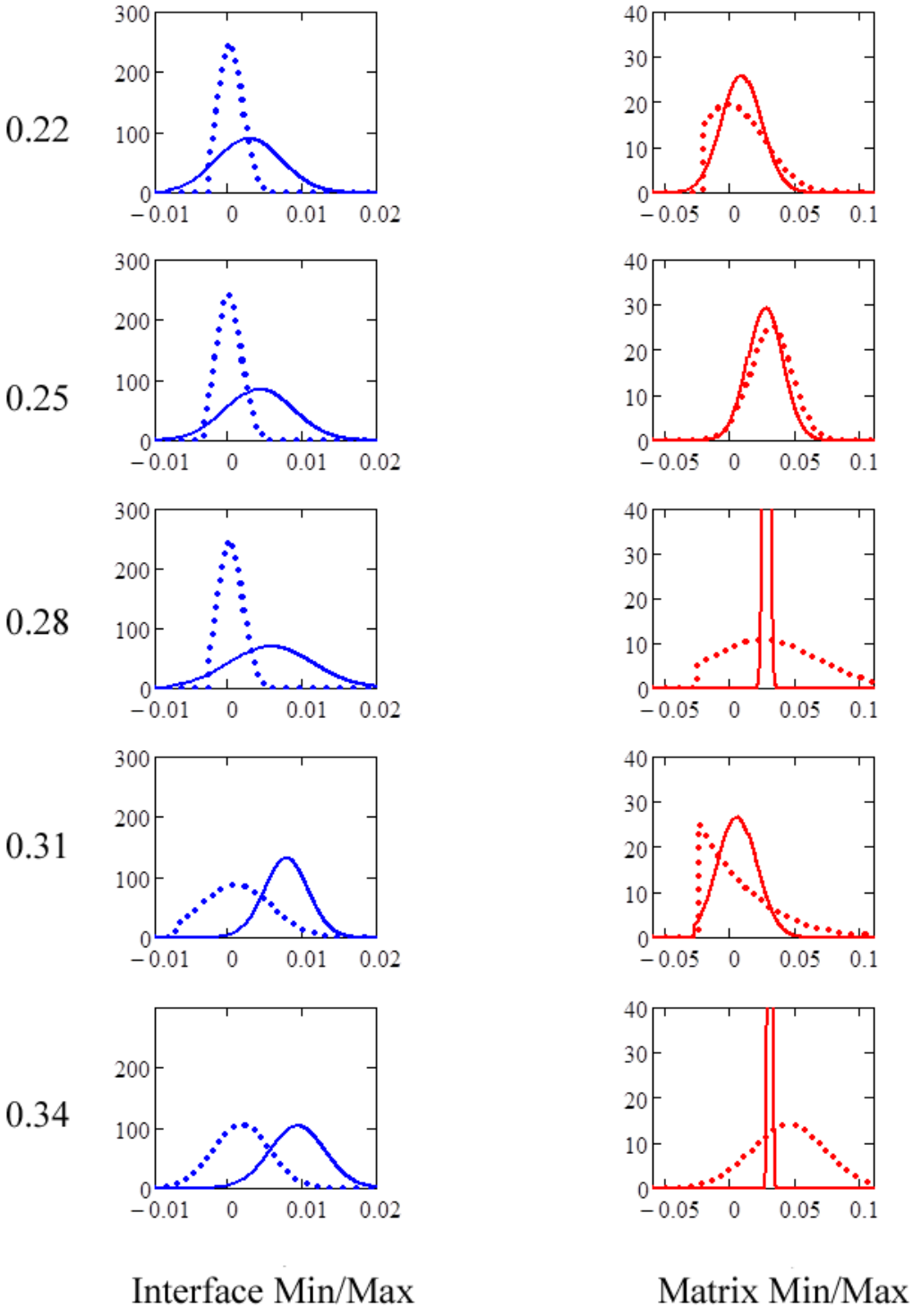

Figure 4.2 Probability distribution function of the strains in the minimum and maximum modulus for various volume fractions around the theoretical percolation threshold.

Chapter 5

Auto Correlation of Local Strain Fields

Autocorrelation is a statistical measure that represents the average covariance whose coordinates differ by a distance vector (k, l)[21]. Spatial autocorrelation arises when the elements of a population, or a data set, are not independent of one another's location in space.[22] Spatial autocorrelation can be used to test the assumption of independence and randomness within a data set, which is critical for many statistical analyses [23]. This work investigates the use of spatial autocorrelation on local mechanical strain fields as a method of identifying and distinguishing between microstructural reinforcing mechanisms in random nanocomposite microstructures. The *xcorr2* function in MATLAB © is modified and used to perform the autocorrelation. Because of the complexity of the microstructures and the difficulty in visualizing the resulting mechanical effects, in this preliminary work, a two dimensional microstructure is studied.

5.1 Autocorrelation in Matlab

MatLab's function *xcorr2* creates a 2-D cross-correlation field between an M by N matrix 'X', and a P by Q matrix '$\bar{H}$'. This is calculated by summing the product of the value of the property at each location, (m, n), and the value of the property located k units away horizontally, and l units away vertically. The formula is given by:

$$C(k,l) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} X(m,n)\bar{H}(m-k, n-l) \tag{5.1}$$

$$-(P-1) \leq k \leq M-1.$$

$$-(Q-1) \leq l \leq N-1.$$

The output matrix, $C(k,l)$, has negative and positive row and column indices. If these values are normalized by dividing the matrix by its largest value, when $k=0,\ \ l=0$, creating a correlogram, r that contains correlation coefficients at each separation distance , (k,l) [21]:

$$r(k,l) = \frac{C(k,l)}{C(0,0)} \tag{5.2}$$

The values in $r(k,l)$ can range from -1 to 1. If $X = \bar{H}$, the corresponding output is the autocorrelation function.

5.2 Boundary Conditions

The challenge to using this function is the presence of boundary error. As the separation distance increases, the analysis is limited because of a lack of data near the boundary of the data set. This results in correlations that decreases radially from the center, regardless of the strength of the correlation at larger distances. This effect can be seen in figure 5.1, which shows an autocorrelation correlogram of a uniform data set, i.e., all values are the same. The correlogram should have a uniform value of one, but because of the boundary condition, it decreases to zero as separation increases. This is easy to see with such a simple example, but the problem becomes more difficult to handle as the complexity of the data set increases.

The goal in this work was to analyze spatial relationships of strains over the entire strain field based on a repeating unit cell. For this model, the data set is a 528 by 528 matrix. This corresponds to an RUC of 48×48 subcells, and strains interpolated at eleven points within each subcell. The error becomes significant when the separation distance is greater than a few subcells in magnitude.

HFGMC imposes periodic boundary conditions, meaning that the strain field

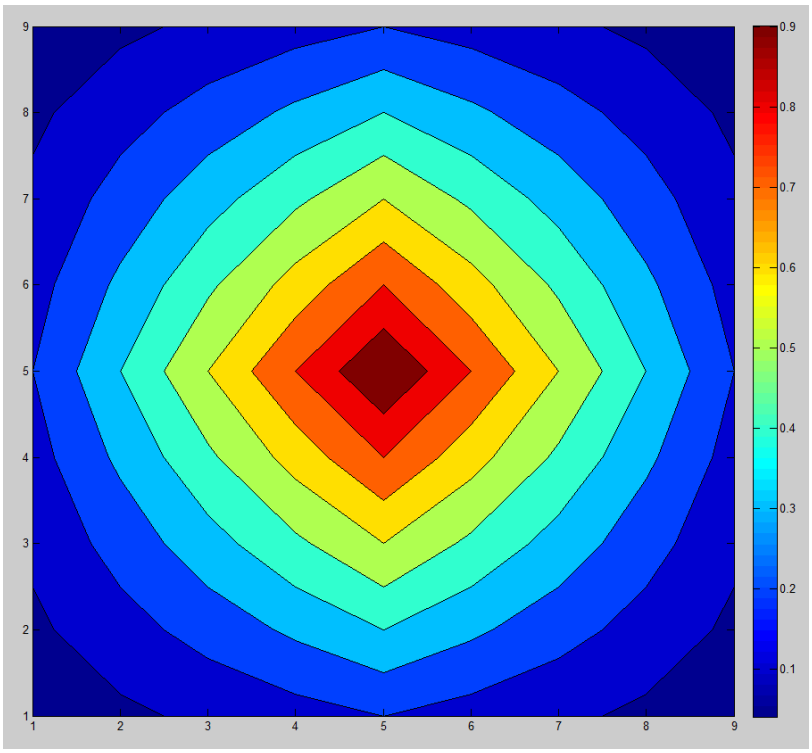

Figure 5.1 A demonstration of how the correlation coefficient diminishes as separation distance increases.

actually represents a doubly periodic array of strains. If $\bar{H}$ is expanded as,

$$\bar{H} = \begin{bmatrix} [X] & [X] & [X] \\ [X] & [X] & [X] \\ [X] & [X] & [X] \end{bmatrix} \tag{5.3}$$

Additional data for the autocorrelation function is available. The error around the boundary still exists, but the extra data shifts the error outside of the original strain field. Figures 5.2-5.4 illustrates how this method affects the results on an actual strain data set.

Figure 5.2 shows a plot of the correlogram of a sample strain field. The plot takes a value of one at the center, which is expected in all normalized autocorrelations. The X and Y axes, which run from 0 to 1152, are mapped to separation distances that run from, -528 to 528. When the separation distance of zero ($k = 0$, $l = 0$) the correlation coefficient takes a value of one at the center of each plot. An area the

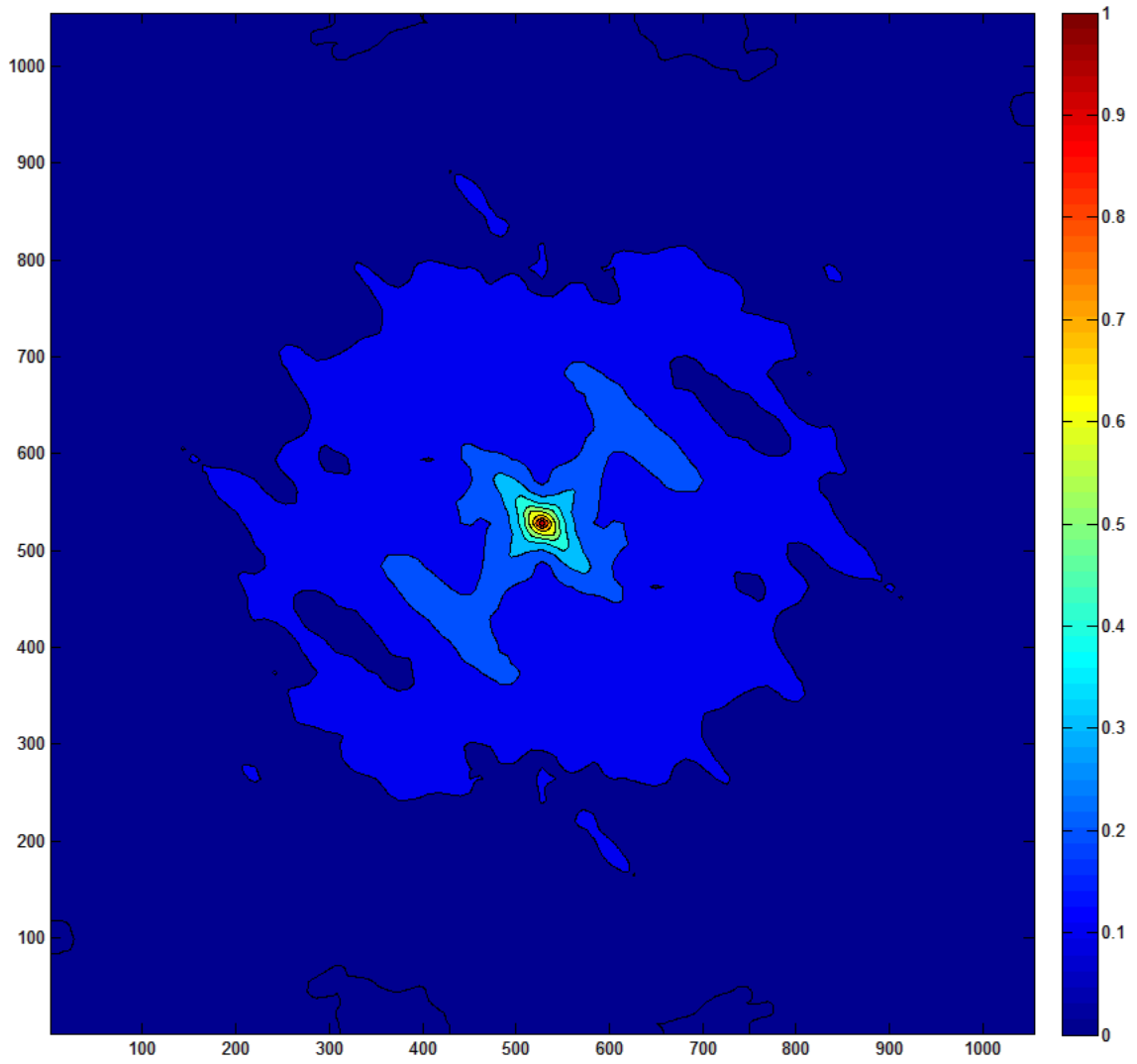

Figure 5.2 Correlogram of a sample strain data set analyzed without periodic boundary conditions

size of one HFGMC subcell around the center shows a correlation coefficient between 0.3 and 0.9, indicating some spatial relationship of strains within one subcell. As the separation distance increases, the correlation coefficient drops from 0 to 0.2, indicating almost no spatial relationship. The low values towards the border of the plot are due to the boundary error.

Figure 5.3 shows an autocorrelation plot, for the same sample strain field, when periodic boundary conditions are applied in *xcorr2*. There are nine points of perfect

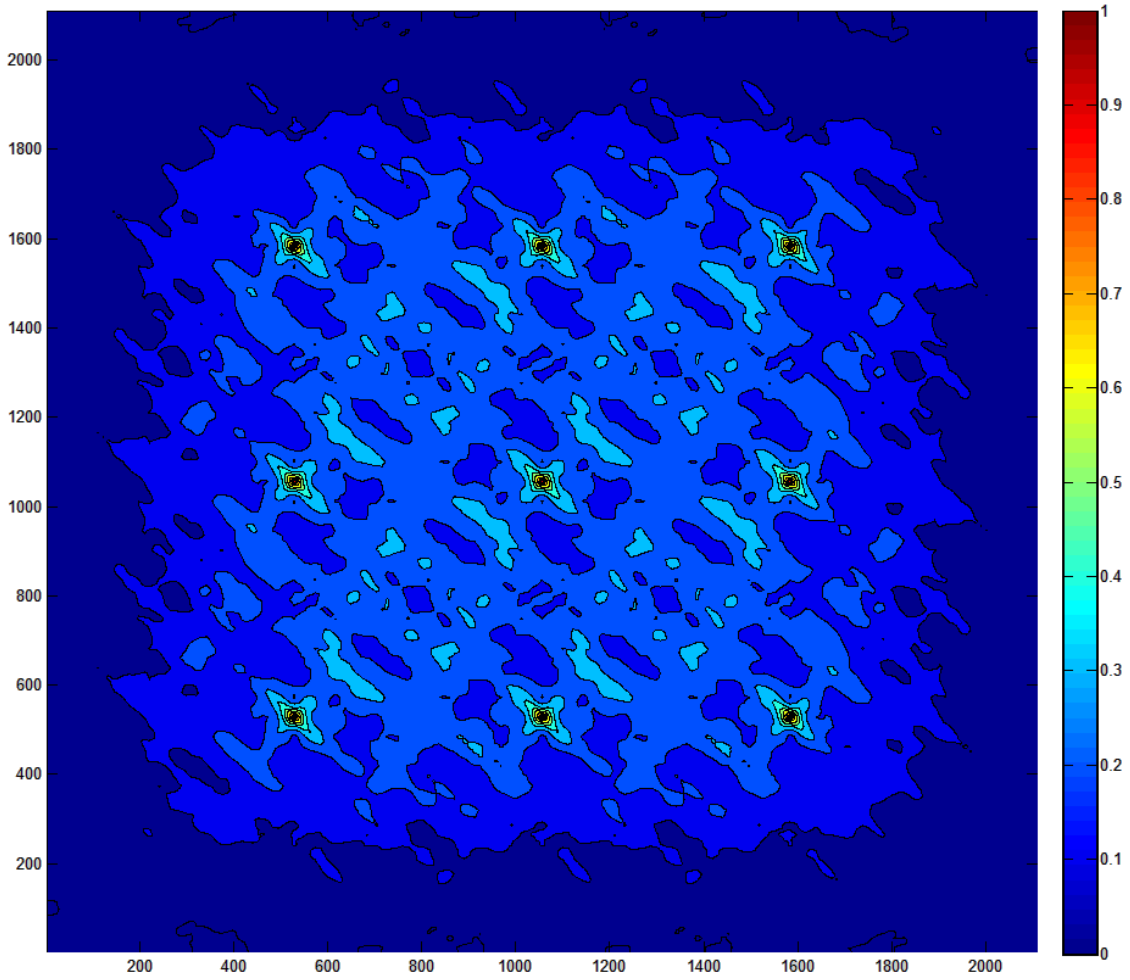

Figure 5.3 The correlogram of the same strain data set analysed with periodic boundary conditions

correlation, which corresponds to the separation distance that matches the periodic boundary conditions exactly. This is expected due to the way the strain fields were replicated. It is also apparent that there is still some error around the boundary of the plot. However, the spatial relationships of the original field can be obtained by extracting the center of this image. This is shown in fig. 5.4. This figure predicts a correlation coefficient field of the same data set from fig. 5.2. By using periodic boundary conditions, more detail is captured towards the edges of the field. The correlation field in fig. 5.4 shows a higher degree of spatial autocorrelation than was predicted in fig. 5.2.

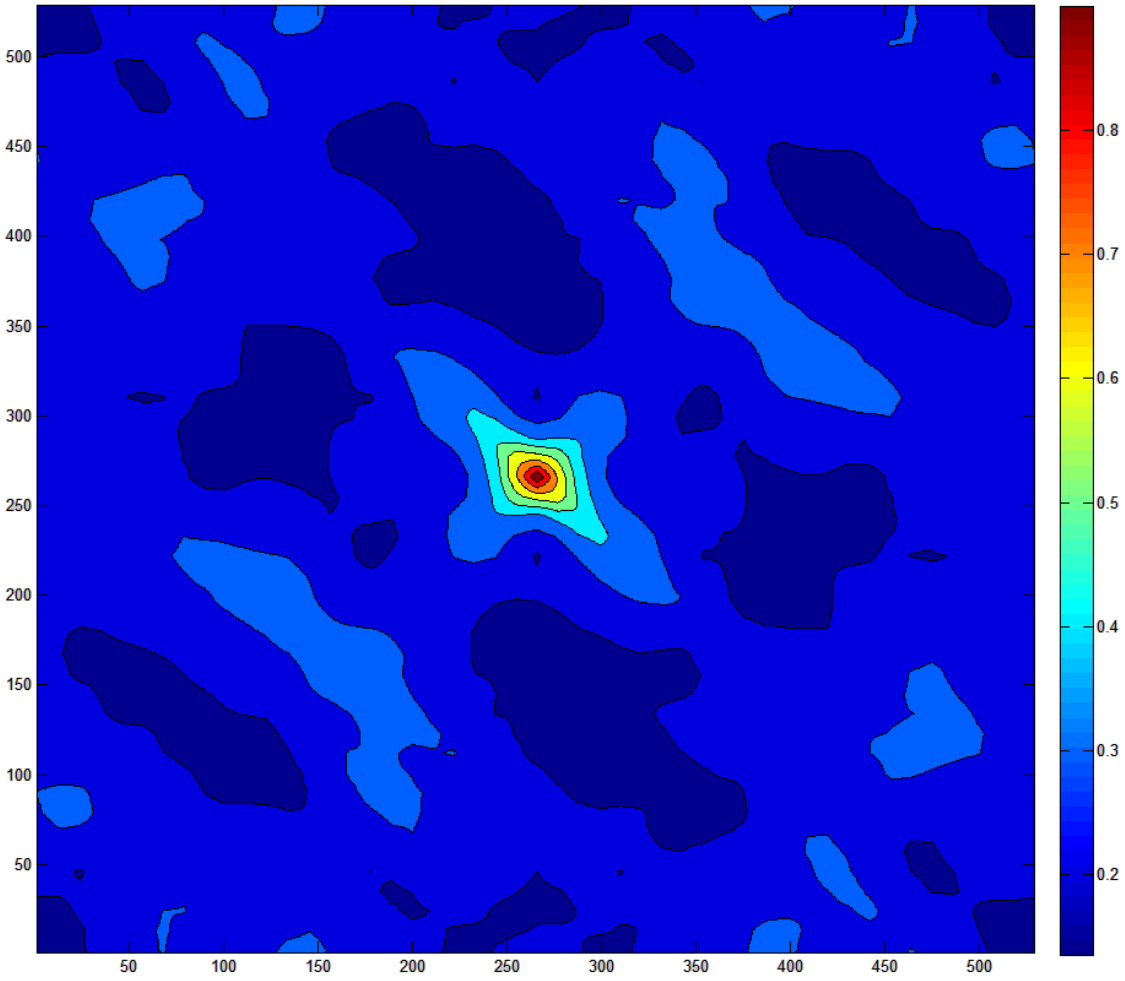

Figure 5.4 The correlogram of the same strain data set trimmed to the size of the original strain field

Results

Corellograms of spatial autocorrelation functions were created using the strain fields from select microstructures. These corellograms can be extremely complex, because the magnitude of each point in the plot, and the relative magnitude of each point to another is important in establishing spatial relationships within a data field. In order to establish some framework to analyse these plots, it is necessary to point out two critical attributes of the corellogram: the baseline value, and local peak values.

Baseline values in this study refer to the average value of the corellogram. This value indicates how homogenous the strain data field is. Low baseline values indicate larger variations in the strain fields in general. A higher baseline value indicates a tendancy for strains to be uniformly distributed throughout the microstructure. A corellogram with a minimum value of 1 (i.e. all values are 1), would indicate a single valued strain field. This is seen in ideal isotropic materials in tension. As gradients appear in the strain field, the baseline value will begin to decrease. The more randomly distributed the strain fields are, the lower the baseline value will be. The frequency that the baseline value occurs in a plot is also important in analysing the spatial relationships. The less frequent the baseline value appears, the more homogeneous the data set will be.

Local peak values are measured relative to the baseline value. The presence of a peak value indicates that a repeating exists in a data set. The strength of the relationship appears to be dependent on the difference between the peak value and the baseline value, rather than the magnitude of the peak on its own. This is due to the fact that this is an averaged effect.

It is important to note that this is an autocorrelation function of the strain fields. These corellograms do not provide a useful tool for analysing the spatial relationships of the constituent materials themselves. While the distribution of strains is effected by the microstructural geometry, other effects are also captured, so the generalization

between strain distribution and material distribution cannot be made. The presence of homogeneous strain fields does not indicate homogeneous distribution of the particles, matrix, or inteface. This holds true for clustering, or any spatial pattern seen in the strains.

Corellograms were created for each volume fraction between 0.05 and 0.90 in increments of 0.05. A few additional corellograms were created around the apparent percolation threshold, at 0.22, 0.28, 0.31, and 0.34 volume fraction. These corellograms contain the spatial autocorrelation function of the strain field for the maximum modulus microstructure at each of these volume fractions.

Figure 5.5 shows the corellograms for volume fractions 0.05, 0.10, 0.15, and 0.20. The 0.05 volume fraction corellogram shows a high baseline value, and this value occupies a very small portion of the plot. As the volume fraction increases up to a volume fraction of 0.20, the baseline value decreases, and the lower correlation values occupy a more significant proportion of the plot. This indicates that as the volume fraction increases in this range, the strains become more randomly distributed within the microstructure.

Figure 5.6 shows the corellograms for volume fractions 0.22, 0.25, 0.28, and 0.30. At the volume fraction 0.22, there is a rise in the baseline value compared to 0.20 in fig. 5.5. Between 0.22 and 0.30, there are large variations in the overall magnitude of the corellograms, and the occurance of spatial patterning is inconsistent.

Figure 5.7 shows the corellograms for volume fractions 0.31, 0.34, 0.35, and 0.40. Volume fraction 0.31 shows a rise in the baseline value compared to the plots seen in fig. 5.6, and there is a high degree of patterning in volume fractions 0.31, and 0.34. Above the volume fraction of 0.31, the baseline values begin to decrease again. This continues in fig. 5.8 and fig. 5.9, until a volume fraction of 0.75 is reached. Above 0.75, there is an increase in the baseline values, and an increase in peak frequency.

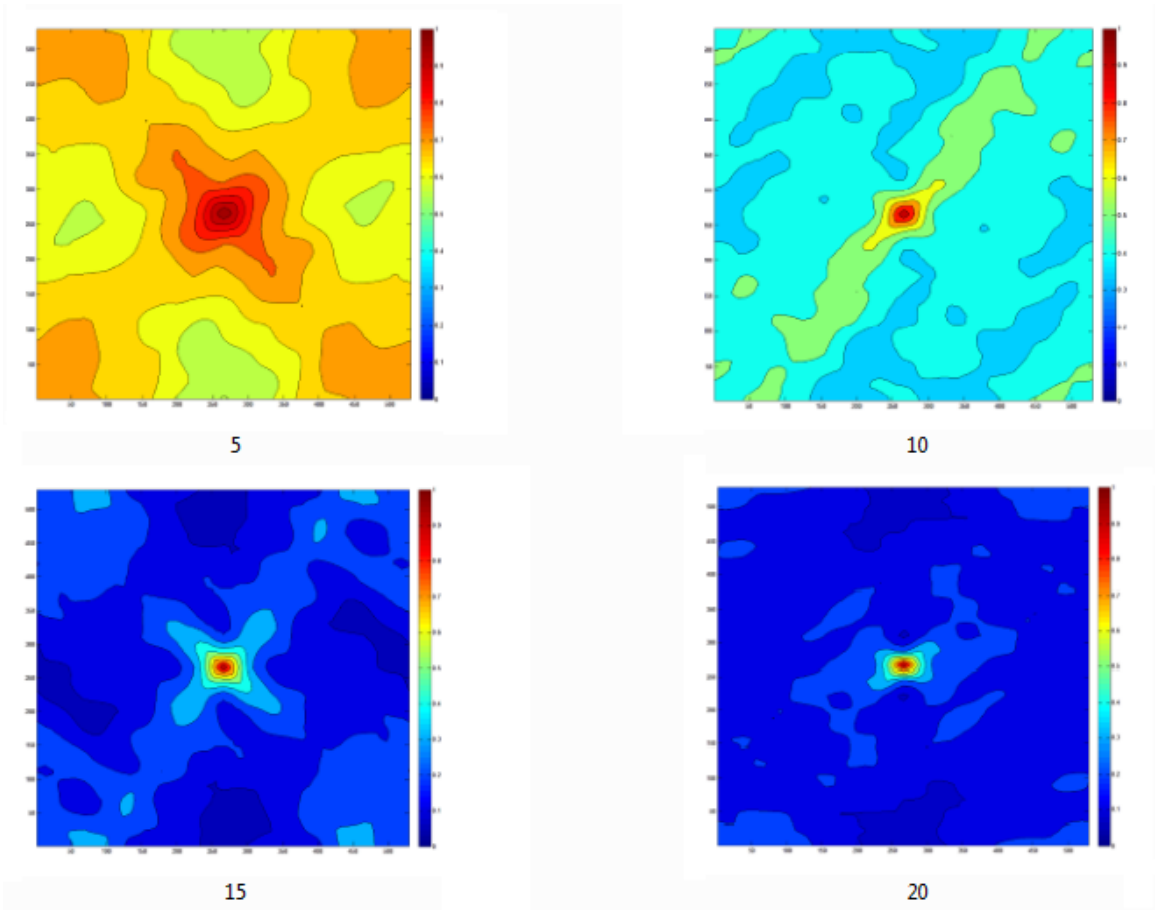

Figure 5.5 Corellelograms of strain fields in the maximum effective modulus microstructures for particle volume fraction 0.05 to 0.20

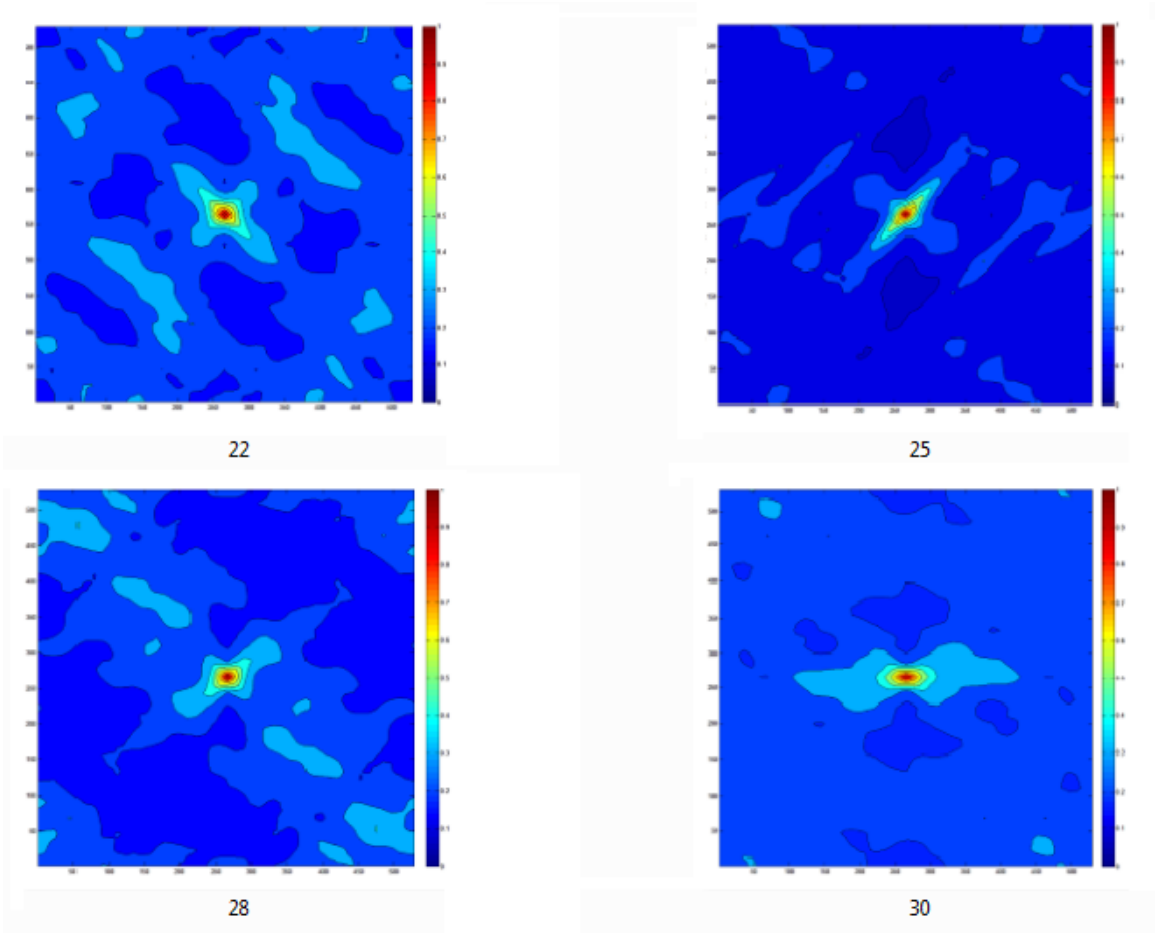

Figure 5.6 Corellelograms of strain fields in the maximum effective modulus microstructures for particle volume fraction 0.22 to 0.30

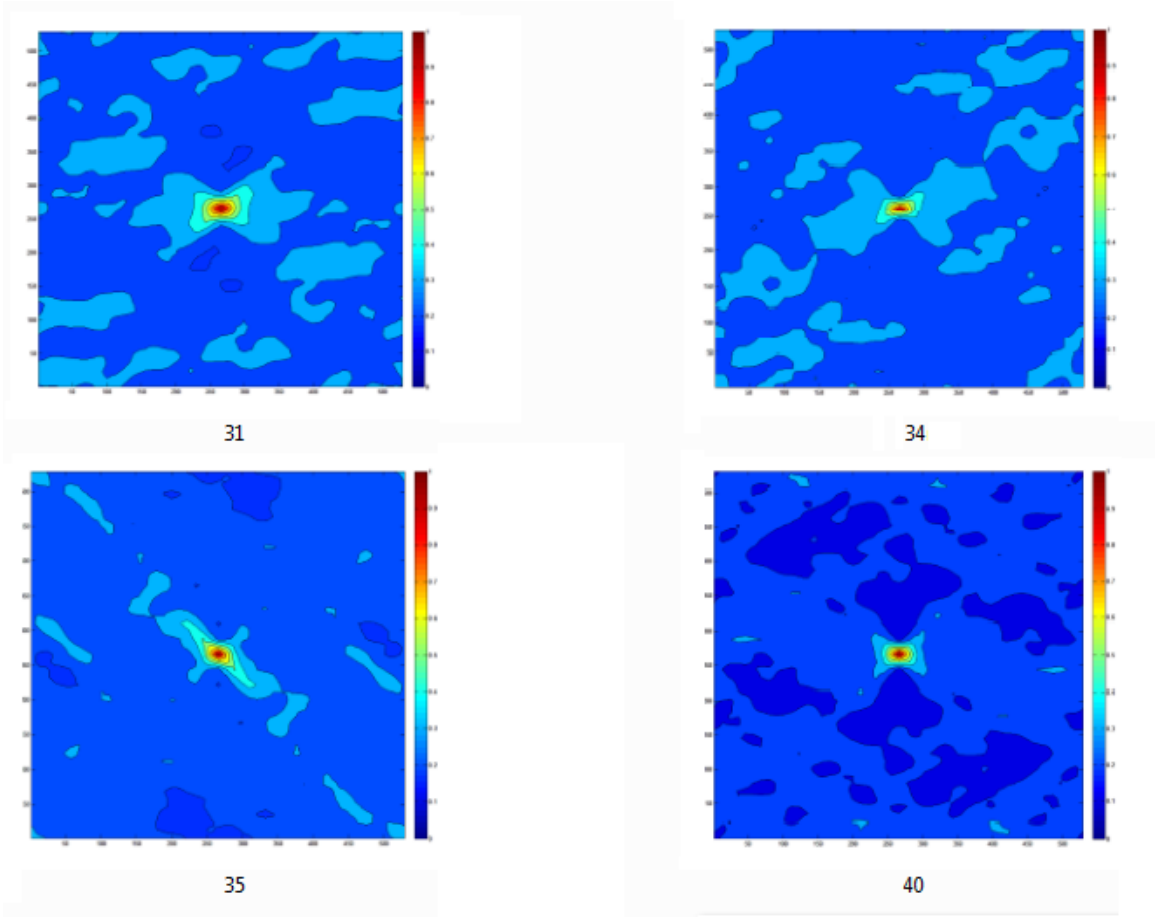

Figure 5.7 Corellelograms of strain fields in the maximum effective modulus microstructures for particle volume fraction 0.31 to 0.40

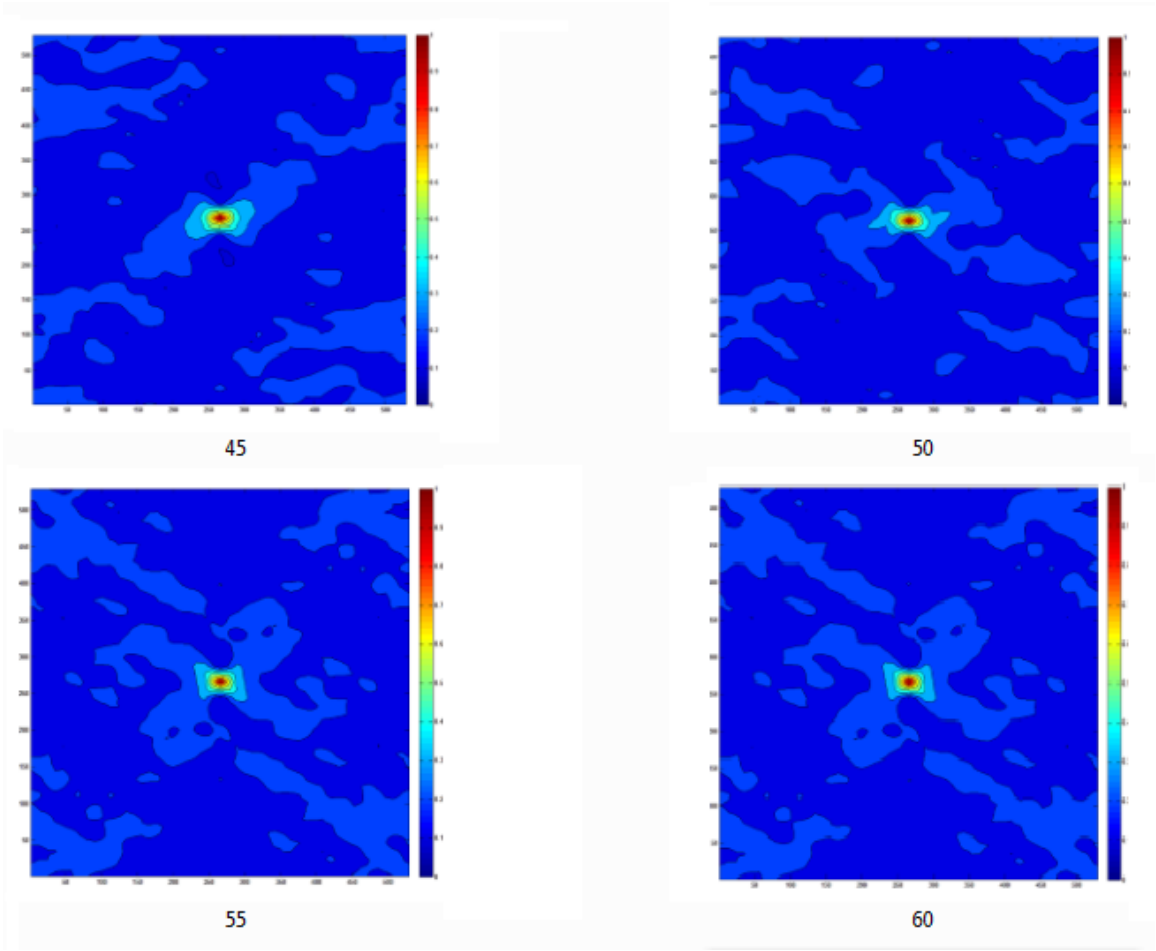

Figure 5.8 Corellelograms of strain fields in the maximum effective modulus microstructures for particle volume fraction 0.45 to 0.60

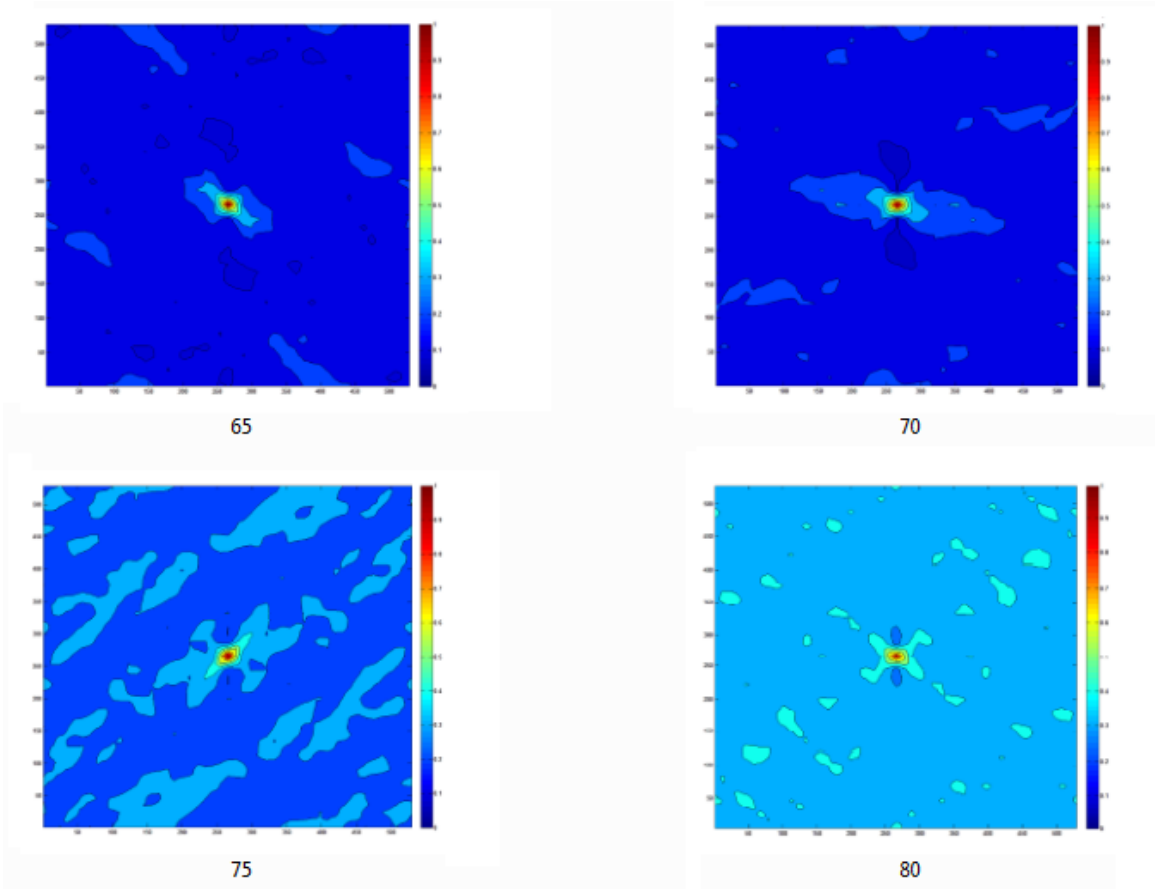

Figure 5.9 Corellelograms of strain fields in the maximum effective modulus microstructures for particle volume fraction 0.65 to 0.80

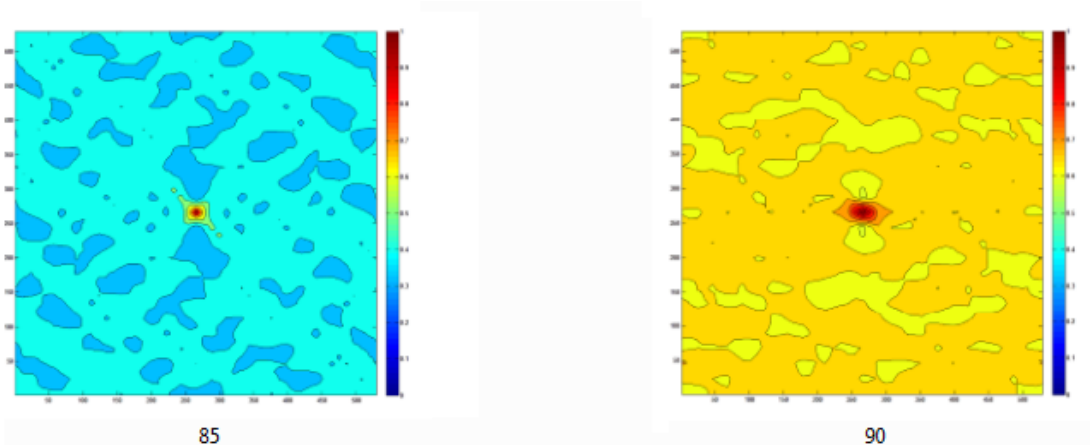

Figure 5.10 Corellelograms of strain fields in the maximum effective modulus microstructures for particle volume fraction 0.85 to 0.90

Based on the apparent percolation threshold determined in Chapter 3, two microstructures were chosen from each of three volume fractions, 0.25, 0.28, and 0.31. One microstructure produced the maximum modulus, and the second produced the minimum modulus at that volume fraction. A comparison of the autocorrelation plots for these volume fraction's minimum and maximum volume fractions are shown in fig. 5.11.

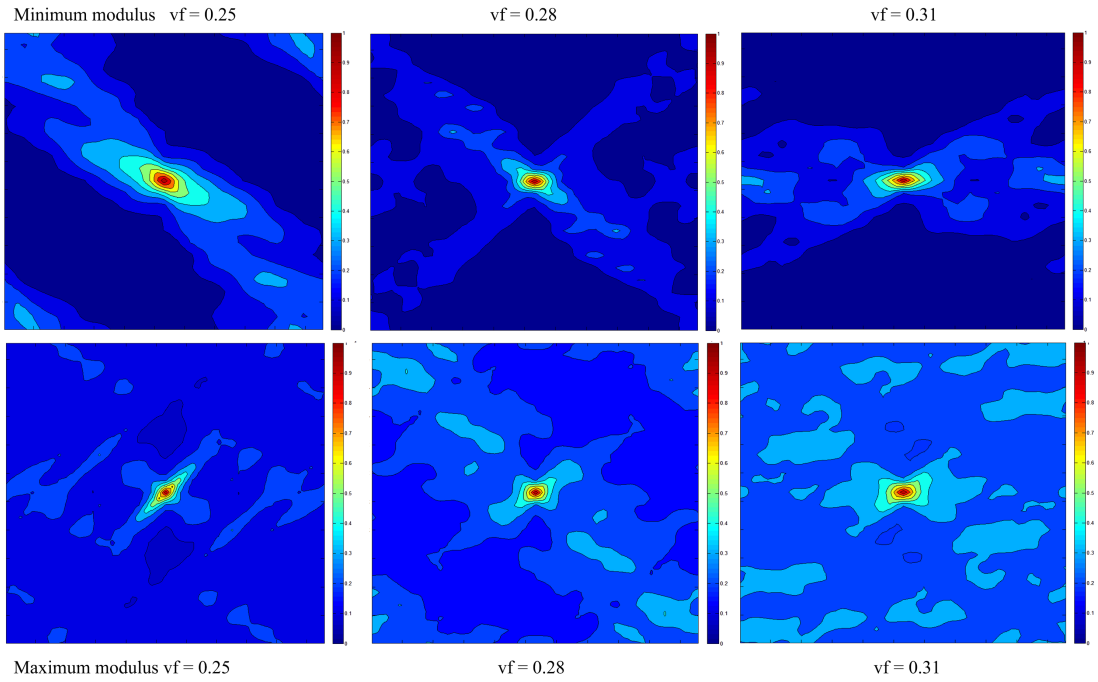

Figure 5.11 Autocorrelations of strain fields of interest. Top row: based on minimum modulus microstructures. Bottom row: based on maximum modulus microstructures.

High correlations close to the center of the plot may indicate some level of clustering of strain values. High correlations separated by regions of low correlation, such as that shown in the maximum vf=0.31, may indicate regularly spaced patterning in the strain field. Both sets display patterning in the correlation values which suggest that the strain fields are not random. Higher correlation coefficients are seen distributed throughout the strain fields for the maximum modulus microstructures than for those from the minimum modulus microstructures. This suggests that the strains are more homogenously in the maximum microstructures than in the minimum.

Chapter 6

Conclusions

Enhanced mechanical properties have been experimentally observed in polymer nanocomposites at uncharacteristically low volume fractions. Mathematical models based on percolation theory provide little insight into the underlying micromechanics. In particular they include neither the effects of the formation of an interface region between the matrix and included phase, which are significant at the nanoscale, nor the influence of complex microstructures on the mechanics response of the composite.

In this work the effective composite properties of a population of simulated two-dimensional random microstructures were calculated. The random microstructures modeled a polymer nanocomposite with three phases, particle, interface and matrix; the interface region surrounds each particle. Statistically, the effective properties of the simulated microstructures predicted percolation-like effects at low volume fractions. In contrast to traditional models, this modeling approach predicts percolation based on the underlying micromechanics rather than a previously defined percolation threshold. Of particular interest were the distributions of properties that developed in the simulations at volume fractions in the region of the percolation-like effects. The volume fractions were, in general, too low for particle phase percolation and very few of the microstructures exhibited connectivity of the combined particle and interface phases. Therefore, in most cases, the increase in stiffness was not due to the formation of a simply connected microstructure.

The goal of this work was to develop methods of identifying and characterizing changes in complex microstructures that could be associated with the distribution

of properties in the simulated microstructures and linked to micromechanical effects. Local strain fields in each phase, and global strain fields of the microstructures were used a a measure of changes and trends in mechanical activity. Two modeling approaches were used. First probability density functions of the strains were constructed using the Principle of Maximum Entropy, and second, spatial autocorrelation fields were calculated for the strain fields of selected microstructures.

The statistical analysis of the strain fields suggests several different types of mechanical mechanisms.

1. At low volume fractions it is likely that a significant portion of the increase in stiffness is due to the addition of a third phase with greater stiffness than the matrix. Regions of interface are concentrated around the particles, and interface-particle connections keep interface strains small. Areas of matrix can be near or far from interface regions; therefore the matrix is less consistently constrained in deformation.

   At the lowest volume fraction presented here, 0.22, the autocorrelation plot shows a wide range of local correlation. This may be due to good distribution of a low volume fraction of particles coupled with large, roughly equal, volume fractions of interface and matrix.

2. At a particle volume fraction of about 0.28, the distribution of elastic moduli moves toward symmetric and the range of matrix strains narrows. This suggests that fewer regions of the matrix are isolated from regions of interface or particles and so they are more uniformly reinforced by the stiffer phases. This suggests that microstructure plays a more significant role in addition to the contribution of the stiffer interface. This volume fraction might serve as a

threshold for the apparent percolation effects - effects due to the formation of particular microstructures.

3. Above particle volume fractions of 0.28, the volume fraction of interface levels off and eventually begins to decrease. At this stage, additional particle subcells are as likely to replace interface subcells as they are to replace matrix subcells. The volume fraction of particles and interface are approaching equal values. Microstructure continues to play a role as regions of matrix are now widely dispersed and are reinforced by the stiffer phases.

4. Autocorrelation fields show similar trends. At low particle volume fractions, particles are widely dispersed and there are large regions of matrix. Each large region of pure matrix, would generate higher correlations over a wider range even in the minimum modulus microstructures.

5. The correlation fields for the maximum modulus microstructures show higher spatial correlation than the minimum modulus microstructures. They also have discontinuous regions of correlation which can be more directly linked to spatially patterned microstructures. The more continuous correlation fields in the minimum microstructures suggest more local similarity.

The mechanisms that produce percolation-like effects at low volume fractions in nanocomposites are more complex than just the formation of connected microstructures. Understanding these mechanisms requires the development of predictive micromechanical models and analysis techniques. These models need to include parameters linked to specific mechanics-based mechanisms, for example, the properties and geometry of interfacial regions, and provide a method for analysis that can capture

the underlying mechanisms, i.e., within a probabilistic framework. This work has demonstrated a modeling framework that can be used to predict low volume fraction percolation effects. Analysis of the model outputs was done using a novel approach that tracks probabilistic changes in local strain fields, rather than local microstructures. This analysis suggests that the effective properties of the composite are likely the result of a sequence of overlapping mechanisms; the influence on the composite of a significant volume fractions of a stiff interface, the contribution of the interface to the formation of a reinforcing/supporting microstructure, and the replacement of matrix by interface, (conversion of all matrix material into interface).

Bibliography

[1] Celzard, A. McRae, E, Deleuze, C., Dufort, M., Furdin, G. and Mâreché, J.F. “Critical concentration in percolating systems containing a high-aspect-ratio-filler", Physical Review B, 53(10), 6209-6214, 1996.

[2] Wu J. McLachlan DS. "Percolation exponents and thresholds obtained from the nearly ideal continuum percolation system graphite-boron nitride." Phys Rev B 1997;53(3);1236-48.

[3] Flandin L. Brechet Y. Canova GR. Cavaille J-Y. "AC electrical properties as a sensor of the microstructural evolution in nanocomposite materials: experiment and simulation." Modeling Simul Mater Sci Eng 1997;7:865-74.

[4] Bethany S. Fralick, Edward P. Gatzke, Sarah C. Baxter, "Three-dimensional evolution of Mechanical Percolation in Nanocomposites With Random Microstructures" Probabilistic Engineering Mechanics 30(2012) 1-8.

[5] Baxter, S.C. and Robinson, C.T. “Pseudo-percolation: Critical volume fractions and mechanical percolation in polymer nanocomposites” Composites Science and Technology, 71, 1273-1279, 2011.

[6] Oskouyi AB, Mertiny P. "Monte Carlo model of the study of percolative thresholds in composites filled with circular conductive nano-disks." Procedia Eng 2011;10:403-8

[7] Ouali, N. Cavaille, J.Y., and Perez, J. "Elastic viscoelastic and plastic behavior of multiphase polymer blends" Plastics, Rubber and Composites Processing and Application, 16, 55-60, 1991.

[8] Favier, V., Chanzy, H. and Cavaille, J-Y. "Polymer Nanocomposites reinforced by Cellulose Whiskers" Macromolecules, 28, 6365-6367, 1995.

[9] McLachlan DS, Blaszkiewicz M, Newnham RE. “Electrical-Resistivity of Composites” J Am Ceram Soc, 73:2178, 1990.

[10] Hashin, Z. (1990) Thermoelastic properties and Conductivity of Carbon/Carbon Fiber Composites, Mechanics of Materials, 8:293-308.

[11] Qiao, R. and Brinson, L.C. "Simulation of interphase percolation and gradients in polymer nanocomposites" Composites Science and Technology, 69, 491-499, 2009.

[12] Niklaus, M, and Shea, H.R., "Electrical conductivity and Young's modulus of flexible nanocomposites made by metal-ion implantation of polydimethylsiloxane: The relationship between nanostructure and macroscopic properties" Acta Materialia, 59, 830-840, 2011.

[13] Paley, M., and Aboudi, J. "Micromechanical analysis of composites by the generalized cells models" Mech. Mat., 14, 127-139, 1992.

[14] Aboudi, Jacob, Steven M. Arnold, and Brett A. Bednarcyk. Micromechanics of composite materials: a generalized multiscale analysis approach. Butterworth-Heinemann, 2012.

[15] Generalized Method of Cells Handbook, TM-2002-212077-VOL3 (examples), Steven M Arnold, Brett A. Bednarcyk, NASA Glenn Research Center, Cleveland, Ohio.

[16] Christensen, R. M. (1979). Mechanics of Composite MAterials. New York, NY: John Wiley and Sons, Inc.

[17] Mori T. and Tanaka K. "Average stress in matrix and average elastic energy of materials with mis-fitting inclusions" Acta Mettalurgia, 2, 571-4, 1973.

[18] Jaynes, E.T. 2003. 'Probability Theory: The Logic of Science Ed. G. Larry Brethorst, Cambridge University Press.

[19] E.T. Jaynes, Where do we stand on maximum entropy? in: R.E. Levine, M. Tribus (Eds.), Maximum Entropy Formalism, MIT Press, Cambridge, MA, 1989, pp. 15-118.

[20] Beltzer, Abraham I., and Tadanobu Sato. "Probability distribution of wave velocity in heterogeneous media due to random phase configuration." Wave motion 38.3 (2003): 221-227.

[21] Ripley, B. D. (1981) Spatial Statistics. Wiley, 78-80pp.

[22] Le Rest, KÃĬvin, David Pinaud, and Vincent Bretagnolle. "Accounting for spatial autocorrelation from model selection to statistical inference: Application to a national survey of a diurnal raptor." Ecological Informatics 14 (2013): 17-24.

[23] Patuelli, Roberto, et al. "The use of spatial filtering techniques: the spatial and space-time structure of German unemployment data." Available at SSRN 893540 (2006).

Appendix A

Matlab RUC Generation Code

```
% This code generates randomized microstructures that are compatable ...
   with HFGMC.
% It has a section that sends the input files to a cloud to decrease ...
   computational cost.
 %It will also pull out relevant results from the output files.

printlevel=1;   % Printlevels.  0=limited debug info, 1=more, -1 ...
   totally off

inputfile='HFGMC_initial.mac';
casestr='HFGMC Initial Attempt';
interface=0;         %interface subcell thickness, will be 0 for ...
   initial tests
BB=1;  GG=1; %dimensions of particle
b=24;  g=24;  %dimensions of RUC, dimensions must be square for 2x ...
   periodic HFGMC
runs=10;

clear E11S N12S E22S N23S E33S G23S G13S G12S NUM  inputfilelist
count=0;

VOLFRACS=[0:0.05:1];
%   VOLFRACS=[VOLFRACS .525 .5725];
volfracs=length(VOLFRACS);

%%  INPUT FILE CONSTRUCTION
% This section reads the template file, creates an input file based
% on current properties with random microstructure.
% inputrandom0.mac has the first volume fraction value, inputrandom1.mac
% has the second volume fraction value, etc. until the max vf is reached
% to complete one run.  Then it starts over at the lowest vf and begins
% the iterations through each vf again to make run two.  And ...
   continues...

for ii=1:runs  %

    for i=1:volfracs;
        if (printlevel>=0)
            disp(' ');
            disp(['ITERATION NUMBER: ' num2str(i) ' of run ' ...
```

```
        num2str(ii) ' of ' ...
         num2str(runs) ' runs, ' num2str(count+1) ' of ' ...
            num2str(runs*volfracs)]);
    disp(' ');
end

newinputfile=[ 'inputrandom' num2str(count) '.mac'];  % Make ...
   file name based on variable i value
inputfilelist{count+1}=newinputfile;
count=count+1;

perc=VOLFRACS(i)
num=floor(perc*(b*g)/(BB*GG)); % exact number of particles
P(i,1)=num*(BB*GG)/(b*g);          % New percentage
NUM(i,1)=num;
M=makematrix_HFGMC(num,b,g,BB,GG,interface);
MM{i}=M;

if (printlevel>=0)
    disp(['Expected number of particles: ' ...
        num2str(b*g*perc/(BB*GG),'%5f') ])
    %disp(['Expected number of particles: ' ...
        num2str(N*N*N*perc/(C*C*C),'%5f') ])
    disp(['Actual number of particles:   ' ...
        num2str(sum(sum(sum(M==1)))/(BB*GG),'%5f') ])
    %disp(['Actual number of particles:   ' ...
        num2str(sum(sum(sum(M==1)))/(C*C*C),'%5f') ])
end

fid1=fopen(inputfile,'r');
fid2=fopen(newinputfile,'w+');
j=0;
while 1 && (j<14)
    j=j+1;
    tline = fgetl(fid1); % Get a line
    if ~ischar(tline), break, end
    %disp(tline);
    fprintf(fid2,tline);  % Write a line to new input file
    fprintf(fid2,'\r\n');
end
fprintf(fid2,[' NB=' num2str(b) ' NG=' num2str(g)]);
%fprintf(fid2,[' NB=' num2str(N) ' NG='
%num2str(N)]); %this is for square particles
fprintf(fid2,'\r\n');

% str=['1'];
 %for cc=2:a
  %   str=[str ',1'];
 %end
 %fprintf(fid2,['  D=' str]);
 %fprintf(fid2,'\r\n');

 str=['1'];
 for cc=2:b
```

```
    str=[str ',1'];
end
fprintf(fid2,['  H=' str]);
fprintf(fid2,'\r\n');

str=['1'];
for cc=2:g
    str=[str ',1'];
end
fprintf(fid2,['  L=' str]);
fprintf(fid2,'\r\n');

for k3=1:g %this is for rods
%for k1=1:N this is for square particles
    %for k1=1:a
        tline='  SM=';
        for k2=1:b
            if M(k2,k3)==1
                tline=[tline '1' ','];
            elseif M(k2,k3)==2
                tline=[tline '2' ','];
            elseif M(k2,k3)==3
                tline=[tline '3' ','];
            else
                disp('num2str hardcoded, odd integer used in M')
                return;
            end
        end
        tline(end)=' ';
        fprintf(fid2,tline);  % Write a line to new input file
        fprintf(fid2,'\r\n');
    %end

 %   if k3<a %this is for rods
 %       tline=['# gamma = ' num2str(k3+1) ];
 %       fprintf(fid2,tline);  % Write a line to new input file
 %   end

    % fprintf(fid2,tline);  % Write a line to new input file
    %fprintf(fid2,'\r\n');
end
tline='*MECH';      fprintf(fid2,tline); fprintf(fid2,'\r\n');
tline='LOP=2';      fprintf(fid2,tline); fprintf(fid2,'\r\n');
tline='NPT=2 TI=0.,200. MAG=0.,0.02 MODE=1'; ...
   fprintf(fid2,tline); fprintf(fid2,'\r\n');
%tline='*THERM'; fprintf(fid2.tline); fprintf(fid2,'\r\n');
%tline='NPT=2 TI=0.,200. TEMP=650.,650.'; ...
   fprintf(fid2.tline); fprintf(fid2,'\r\n');
tline='*SOLVER';    fprintf(fid2,tline); fprintf(fid2,'\r\n');
tline='METHOD=1 NPT=2 TI=0.,200. STP=1.'; ...
   fprintf(fid2,tline); fprintf(fid2,'\r\n');
tline='NLEG=5 NINTEG=11'; fprintf(fid2,tline); ...
   fprintf(fid2,'\r\n');
tline='*PRINT'; fprintf(fid2,tline); fprintf(fid2,'\r\n');
```

```
        tline='NPL=6'; fprintf(fid2,tline); fprintf(fid2,'\n\n');
        tline='*END'; fprintf(fid2,tline);  fprintf(fid2,'\r\n');

        fclose(fid1);  % Close template file
        fclose(fid2);  % close new input file

    end
end

%% Run code section
% This section creates and then submits the inputrandom*.mac files to
% condor to be executed.

% First, write condor submit file
fid3=fopen('macsubmitfile','w+');
fprintf(fid3,'universe = vanilla\n');
fprintf(fid3,'requirements =  (Arch == "X86_64") \n');
%fprintf(fid3,'environment = path=c:\\winnt\\system32 \n');
fprintf(fid3,'executable = mac4.exe \n');
fprintf(fid3,'input = inputrandom$(Process).mac\n');
fprintf(fid3,'arguments = inputrandom$(Process)\n');
fprintf(fid3,'output = inputrandstdout$(Process).out\n');
fprintf(fid3,'error = inputrandom$(Process).err\n');
fprintf(fid3,'log = inputrandom$(Process).log\n');
fprintf(fid3,['queue ' num2str(runs*volfracs) ' \n']);
fclose(fid3);  % close new input file

str=['C:\HPC\condor\bin\condor_submit.exe macsubmitfile '];
system(['erase inputrandom*.out']);  % erase out file before condor ...
   starts
if (printlevel>=0)
    disp(' ');
    disp(' Running condor, command is:')
    disp([ '    ' str])
end
tic

[status,result]=system(str);
result

% check to see if the output files are all done.  Could just check
% inputrandom.out and see if it is complete...

d=dir('inputrandom*.out');

running=1;
while running
    pause(1);
    disp([' Checking to see if CONDOR is done...' num2str(length(d)) ...
         ...
        ' output files finished of '  num2str((runs*volfracs)) ]);
    d=dir('inputrandom*.out');
    if (length(d)==(runs*volfracs))
        running=0;
```

```
    end
    if (length(d)>0)
        for ddd=1:length(d)
            if d(ddd).bytes<1000
                running=1;
            end
        end
    end
end

if (printlevel>=0)
    disp([' Finished running executable, time: ' num2str(toc) ' ...
       seconds'])
    disp(' ');
end



%% Read output section
% This section reads throught the output file and looks for a specific
% string.  When that is found, it reads the numeric value and puts the
% value into an array.

count=0;
for ii=1:runs
    for i=1:volfracs;
        if (printlevel>=0)
            disp(' ');
            disp(['ITERATION NUMBER: ' num2str(i) ' of run ' ...
               num2str(ii) ' of ' ...
                num2str(runs) ' runs, ' num2str(count+1) ' of ' ...
                   num2str(runs*volfracs)]);
            disp(' ');
        end

        newinputfile=[ 'inputrandom' num2str(count) '.out']
        outfile=[newinputfile(1:end-4) '.out'];
        count=count+1;

        if length(dir(outfile))>0
            %start case where output file is there
            fid3=fopen(outfile,'r');
            tic;
            while fid3<0
                disp('Attempting to open output file again!')
                fid3=fopen(outfile,'r');
                pause(1)
                if toc>40
                    return
                end
            end %end try to open file

            frewind(fid3);
```

```
if (printlevel>=0) disp('Searching output file for ...
   strings');end
while 1
    tline = fgetl(fid3); % Get a line
    if ~ischar(tline), disp('BREAK'); break, end

    if length(tline)>10
        if strcmp('     E11S=',tline(1:10))
            if (printlevel>=1)
                disp('Found output string:');
                disp(tline);
            end
            E11S(i,ii)=str2num(tline(11:end));  % Store ...
               desired value in array
        end
        if strcmp('     N12S=',tline(1:10))
            if (printlevel>=1)
                disp('Found output string:');
                disp(tline);
            end
            N12S(i,ii)=str2num(tline(11:end));  % Store ...
               desired value in array
        end
        if strcmp('     E22S=',tline(1:10))
            if (printlevel>=1)
                disp('Found output string:');
                disp(tline);
            end
            E22S(i,ii)=str2num(tline(11:end));  % Store ...
               desired value in array
        end
        if strcmp('     N23S=',tline(1:10))
            if (printlevel>=1)
                disp('Found output string:');
                disp(tline);
            end
            N23S(i,ii)=str2num(tline(11:end));  % Store ...
               desired value in array
        end
        if strcmp('     E33S=',tline(1:10))
            if (printlevel>=1)
                disp('Found output string:');
                disp(tline);
            end
            E33S(i,ii)=str2num(tline(11:end));  % Store ...
               desired value in array
        end
        if strcmp('     G23S=',tline(1:10))
            if (printlevel>=1)
                disp('Found output string:');
                disp(tline);
            end
            G23S(i,ii)=str2num(tline(11:end));  % Store ...
               desired value in array
```

```
    end
    if strcmp('     G13S=',tline(1:10))
        if (printlevel>=1)
            disp('Found output string:');
            disp(tline);
        end
        G13S(i,ii)=str2num(tline(11:end));  % Store ...
           desired value in array
    end
    if strcmp('     G12S=',tline(1:10))
        if (printlevel>=1)
            disp('Found output string:');
            disp(tline);
        end
        G12S(i,ii)=str2num(tline(11:end));  % Store ...
           desired value in array
    end


end %end tline>10

if length(tline)>43
    if strcmp('          MATERIAL NO.= 1     VOLUME ...
       RATIO=',tline(1:43))
        if (printlevel>=1)
            disp('Found output string:');
            disp(tline);
        end
        M1VF(i,ii)=str2num(tline(44:end));  % Store ...
           desired value in array
    end


    if strcmp('          MATERIAL NO.= 2     VOLUME ...
       RATIO=',tline(1:43))
        if (printlevel>=1)
            disp('Found output string:');
            disp(tline);
        end
        M2VF(i,ii)=str2num(tline(44:end));  % Store ...
           desired value in array
    end


    if strcmp('          MATERIAL NO.= 3     VOLUME ...
       RATIO=',tline(1:43))
        if (printlevel>=1)
            disp('Found output string:');
            disp(tline);
        end
        M3VF(i,ii)=str2num(tline(44:end));  % Store ...
           desired value in array
    end
```

```
                end %end tline>43

            end %end while loop
            fclose(fid3);

        else %if output file is not there
            vars={'E11S' 'N12S' 'E22S' 'N23S' 'E33S' 'G23S' 'G13S' ...
                'G12S' 'M1VF' 'M2VF' 'M3VF'};
            E11S(i,ii)=-1;
            N12S(i,ii)=-1;
            E22S(i,ii)=-1;
            N23S(i,ii)=-1;
            E33S(i,ii)=-1;
            G23S(i,ii)=-1;
            G13S(i,ii)=-1;
            G12S(i,ii)=-1;
            M1VF(i,ii)=-1;
            M2VF(i,ii)=-1;
            M3VF(i,ii)=-1;
        end %end checking if file is there
    end
end

%the following deletes the extra files created by condor.  not ...
   necessary to
%keep
system(['erase inputrandom*.log']);
system(['erase inputrandom*.err']);
system(['erase inputrandstd*']);
```

Appendix B

Example HFGMC Input Code

```
MAC/GMC 4.0 G: 2—D HIGH FIDELITY GMC
*CONSTITUENTS
  NMATS=3
# —— Gold nanorod units are Pascals —
  M=1 CMOD=6 MATID=U MATDB=1 &
  EL=78.E9,78.E9,0.35,0.35,28.8E9,—0.68E—6,9.74E—6
# —— Polymer Matrix
  M=2 CMOD=6 MATID=U MATDB=1 &
  EL=78.E3,78.E3,0.45,0.45,26.9E3,—0.68E—6,9.74E—6
# —— Interface Layer with geometric mean properties
  M=3 CMOD=6 MATID=U MATDB=1 &
  EL=78E6,78E6,0.45,0.45,26.9E6,—0.68E—6,9.74E—6
*RUC
MOD=22 ARCHID=99
 NB=4 NG=4
  H=1,1,1,1
  L=1,1,1,1
  SM=3,2,2,2
  SM=3,2,2,2
  SM=3,2,2,3
*MECH
LOP=2
NPT=2 TI=0.,200. MAG=0.,.0100 MODE=1
*SOLVER
METHOD=1 NPT=2 TI=0.,200. STP=1.
NLEG=5 NINTEG=11
*PRINT
NPL=6
```