5-2013

Variational Asymptotic Method for Unit Cell Homogenization of Thermomechanical Behavior of Composite Materials

Chong Teng
Utah State University

VARIATIONAL ASYMPTOTIC METHOD FOR UNIT CELL HOMOGENIZATION OF THERMOMECHANICAL BEHAVIOR OF COMPOSITE MATERIALS

by

Chong Teng

A dissertation submitted in partial fulfillment
of the requirements for the degree

of

DOCTOR OF PHILOSOPHY

in

Mechanical Engineering

Approved:

Dr. Wenbin Yu Major Professor	Dr. Thomas H. Fronk Committee Member
Dr. Steven L. Folkman Committee Member	Dr. Ling Liu Committee Member
Dr. Zhaohu Nie Committee Member	Dr. Mark R. McLellan Vice President for Research and Dean of the School of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2013

Abstract

Variational Asymptotic Method for Unit Cell Homogenization of Thermomechanical Behavior of Composite Materials

by

Chong Teng, Doctor of Philosophy

Utah State University, 2013

Major Professor: Dr. Wenbin Yu
Department: Mechanical and Aerospace Engineering

The properties of materials have been investigated throughout the twentieth century. However, with more and more knowledge in material science, it became extremely hard for individual materials to meet every specific requirements of engineering design in this modern world of high efficiency and performance. To fulfill the design needs of engineering structures, composite materials were widely developed in various ways since early $1990s$. This leads to an enormous amount of research in the field of composites; moreover, researchers focused more and more on engineering microstructures in order to improve the performance of composite materials.

Problems of composite materials, which are often observed with complicated geometries, are very difficult to achieve analytical solutions. Therefore, the use of numerical methods such as finite element method (FEM) is required for solving such problems. With the fast development of FEM, this numerical method is well established and recognized by more and more analysts and scientists. This numerical analysis tool is very powerful to obtain behaviors of engineering structures under different boundary conditions and loads. However, for problems of composites featuring heterogeneity, the total degrees of freedom

of the composite materials can be so large that even with the significant strides in computer hardware, the direct finite element analyses of such composites sometimes could be impossible. A microscopic building block (aka unit cell or representative volume element or representative structural element in literature) which stores the necessary local information of composites is used to carry out an analysis in microscopic level in order to obtain the effective material properties, and after that to recover the corresponding local stress and strain fields within the original heterogeneous material based on the global behavior of the macroscopic structural analysis.

The thermomechanical behavior of materials is always concerned in engineering because of the temperature dependent material performance in the nature. Almost all materials under their working conditions cannot be kept at unchanged temperature fields which makes the study of thermomechanical behavior of materials meaningful and important. In this dissertation, micromechanics modeling of such problems are developed based on variational asymptotic method which uses a variational statement to solve such problems. This methodology is more efficient as it only deals with one functional while the traditional asymptotic method deals with a group of differential equations. Variational Asymptotic Method for Unit Cell Homogenization (VAMUCH) has been developed recently and will be used to conduct the micromechanics modeling throughout this dissertation. The following problems will be addressed in this dissertation: (1) micromechanics modeling of composites with temperature dependent constituent properties; (2) micromechanics modeling of composites with finite temperature variations; (3) micromechanics modeling of composites under nonuniformly distributed temperature field; and (4) micromechanics modeling of composites under internal and external loads.

(232 pages)

Public Abstract

Variational Asymptotic Method for Unit Cell Homogenization of Thermomechanical Behavior of Composite Materials

by

Chong Teng, Doctor of Philosophy
Utah State University, 2013

Major Professor: Dr. Wenbin Yu
Department: Mechanical and Aerospace Engineering

To seek better material behaviors, the research of material properties has been massively carried out in both industrial and academic fields throughout the twentieth century. Composite materials are known for their abilities of combining constituent materials in order to fulfill the desirable overall material performance. One of the advantages of composite materials is the adjustment between stiffness and lightness of materials in order to meet the needs of various engineering designs. Even though the finite element analysis is mature, composites are heterogeneous in nature and can present difficulties at the structural level with the acceptable computational time. A way of simplifying such problems is to find a way to connect structural analysis with corresponding analysis of representative microstructure of the material, which is normally called micromechanics modeling or homogenization.

Generally speaking, the goal of homogenization is to predict a precise material behavior by taking into account the information stored in both microscopic and macroscopic levels of the composites. Of special concern to researchers and engineers is the thermomechanical behavior of composite materials since thermal effect is almost everywhere in real practical

cases of engineering. In aerospace engineering, the thermomechanical behaviors of composites are even more important since flight under high speed usually produces a large amount of heat which will cause very high thermal-related deformation and stress.

In this dissertation, the thermomechanical behavior of composites will be studied based on the variational asymptotic method for unit cell homogenization (VAMUCH) which was recently developed as an efficient and accurate micromechanics modeling tool. The theories and equations within the code are based on the variational asymptotic method invented by Prof. Berdichevsky. For problems involving small parameters, the traditional asymptotic method is often applied by solving a system of differential equations while the variational asymptotic method is using a variational statement that only solves one functional of such problems where the traditional asymptotic method may apply.

First, we relax the assumption made by traditional linear thermoelasticity that not only a small overall strain is assumed to be small but also the temperature variation. Of course, in this case we need to add temperature dependent material properties to VAMUCH so that the secant material properties can be calculated. Then, we consider the temperature field to be point-wise different within the microstructure; a micromechanics model with nonuniformly distributed temperature field will be addressed. Finally, the internal and external loads induced energies are considered in order to handle real engineering structures under their working conditions.

To my parents
Xuexiang Teng & Changxin Zhao

Acknowledgments

For years of study to achieve the final goal, without those who helped me and guided me in many ways, I could not have gone anywhere so far. I would like to thank the following people for their helps throughout this period of my life.

Foremost, I am extremely grateful to my advisor, Professor Wenbin Yu, for his consistent guidance, pioneering inspiration, and tireless encouragement during my graduate studies at Utah State University. His professional manners of research, intelligent ways of thinking, and charming personality deeply impressed me and inspired my intellect. He led me into the field of micromechanics and taught me every single knowledge he could in a patient way in order to assist my study during these years. Moreover, he set a phenomenal example as a person and researcher in my life, which I will never forget.

I would also like to thank Thomas H. Fronk, Steven L. Folkman, Ling Liu, and Zhaohu Nie for serving on my dissertation committee and for their dedication in evaluating this dissertation. All the advice and suggestions from them are really appreciated and helped me greatly improve details of the works that have been accomplished.

Also, many thanks go to my academic advisors, Bonnie Ogden, Christine Spall, and Karen Zobell, for their helpful advice and generous care of my student life. Special thanks go to our former and current department heads, Professor Byard Wood and Robert E. Spall, for their suggestions and efforts to help me pass all difficulties in my study.

Thanks for all the precious helps, discussions, and friendships from my colleagues such as Drs. Hui Chen, Qi Wang, Changyong Lee, Zheng Ye, Liang Zhang, and Hemaraju Pollayi, as well as Mr. Changjin Choi, Fang Jiang, and Hamsasew M. Sertse.

I would like to acknowledge Liebherr Aerospace company and the US Air Force Research Laboratory (AFRL) for their financial support for my graduate study. Also I would like to acknowledge the Department of Mechanical and Aerospace Engineering and the School of Graduate Studies of Utah State University for providing the supporting travel funds several times.

To my dear father, Xuexiang Teng, and my dear mother, Changxin Zhao, I could not find such words to precisely describe my love and gratitude to both of them for everything they did and all the moments they shared in my life. I could not have gone anywhere without the company from both of them. Also it is worthy to mention my dear maternal grandfather, Yongyao Zhao, for his guidance, support, and solicitude. Finally, I am deeply indebted to my wife, Di Wang, for her tenderness, love, support, and encouragement to me all the time.

Chong Teng

Contents

List of Tables

List of Figures

Acronyms

VAM	variational asymptotic method
VAMUCH	Variational Asymptotic Method for Unit Cell Homogenization
FEM	finite element method
FEA	finite element analysis
UC	unit cell
RVE	representative volume element
RSE	representative structural element
PDE	partial differential equation
EIAS	equivalent inclusion-average stress
MOC	method of cells
GMC	generalized method of cells
ECM	elasticity-based cell model
UC	unit cell
MPE	principle of minimum potential energy
ODE	ordinary differential equation
PVW	principle of virtual work
IVW	internal virtual work
EVW	external virtual work
CTE	coefficient of thermal expansion

Chapter 1

Introduction

1.1 Background and Motivation

Conventional individual materials started to be used as ingredients of combination of two and more materials in industry since early 1990s in order to fulfill the needs of good overall material properties. This is caused by development of modern technology which induces more and more complicated engineering structures and more and more high-level requirement of material performance. For example, in aerospace engineering, high strength and weight ratio, high abrasion resistance, low thermal expansion, and good thermal insulation are always required and of course since the combination techniques of materials have become available, studies and researches conducted in the field of composite materials became necessary. However, the development of composite materials led to intricate engineering structures such as woven, sandwich or corrugated shaped composites, nanotubes, etc. Though the development in numerical methods, computer software, and hardware benefits engineering analyses and designs, for such engineering structures mentioned above with high heterogeneity and complexity, it is really needed to seek for a more simplified and efficient way to analyze them.

Since continuum mechanics typically applies when the scale of a phenomenon is much larger than the separation between the constituent atoms of the material under consideration, the ingredient materials in composites can be considered using continuum mechanics. By knowing this, we assume the materials are continuously distributed, fulfilled the entire region of their occupied space, and ignores the discontinuities induced by the space between molecules. Structural analysis is used to determine the effects of loads, geometries, and corresponding boundaries related with structural deformation. Structural problems usually

can be solved either by analytical methods or numerical methods. Typical analytical methods such as mechanics of material and elasticity theory are often used to deal with simple cases which are solvable by hand. Finite element method is perhaps the most popular method in structural analysis which is a numerical method to obtain approximate solutions in such analyses. It divides the domain into a system of discrete subdomains (aka elements) with connecting points (aka nodes) and solve the governing equations which are usually partial differential equations (PDEs) by converting those PDEs into an algebraic system using shape functions, and seeking an approximate solution numerically.

In engineering problems of composite materials, finite element analysis is commonly used in industry for many years. However, for complex structures and heterogeneous materials, to carefully model them (minimize the error) using FEA, we need to have fine mesh in every part of the structure and material which results a massive number of degrees of freedom. It is very easy to make the whole problem very expensive or inefficient using the existing computational resources. A natural way to overcome this kind of difficulties is to replace the original composites with a large number of heterogeneity with an equivalent homogeneous model of composites. This is a special case of micromechanics called homogenization in the area of applied mechanics which is referred as a scientific discipline to study the response of heterogeneous materials by treating them as formed by homogeneous materials with effective material properties. The benefit of homogenization is to dramatically reduce the global degrees of freedom of the structures as well as maintain reasonable accuracy in order to approximate the original analyses using more efficient, economic, and simpler ones.

Homogenization is usually considered to be accomplished by three steps as illustrate in Fig. 1.1. This takes advantage of a microscopic building block (aka unit cell or representative volume element or representative structural element in literature). The first step of homogenization is to embed the original problem into similar problems by introducing the small parameter "ε", then the effective properties of the microscopic building block can be obtained by a micromechanical analysis. The second step is to carry out the macroscopic

analysis of the structure with the heterogeneous materials replaced with imaginary homogeneous materials with the effective properties we just obtained. The final step is to recover the local fields within the orignal heterogeneous materials based on the global behavior of the macroscopic structural analysis.

To carry out homogenization, identifying a microscopic building block is always important and inevitable. This kind of microscopic building block has many definitions in literature. For example, unit cell is widely used in crystal structure of material which is defined in terms of its lattice points and within each unit cell is the smallest unit of the crystal that the material can be divided into. Representative volume element is also often mentioned in micromechanics of composites which usually stands for the smallest material volume element that sufficient material information is stored to represent the mean response of the whole material.

The recent research of homogenization brought us another concept of this microscopic building block which is called representative structural element (RSE). The major difference between RSE and RVE is that RSE uses the lowest dimension possible to describe the heterogeneity while RVE's dimensionality is defined by both analysis requirement and heterogeneity. In another word, for a fiber reinforced composites, if 3D properties are needed, 3D RVE will be chosen while only a 2D RSE will be needed in this case. The concept of RSE fills the gap between structural mechanics and micromechanics by considering structural mechanics as a special case of micromechanics. Furthermore, using certain method such as variational asymptotic method, for any structures with periodicity in one or more dimensions such as fiber reinforced composites, a complete set of 3D effective material properties will be obtained even with a RSE of lower dimension. This will of course save a lot of computational efforts and make the analyses more efficient.

In industry, one of the obstacles of using composite materials is the failure or damage caused by local thermal stresses due to thermal loads or mismatching CTEs between constituent materials. Especially in applications of aerospace engineering, almost all of them need to undergo thermal loadings under their working conditions. As the heterogeneous

and complicated structures started to be applied in this field, more and more concerns are generated in the field of micromechanics modeling in order to carry out efficient yet accurate analyses which make the study in this dissertation very important and interesting. However, in many approaches in micromechanics, it barely appears to have any approaches that taking account of temperature dependent material properties, mechanical and thermal loads coupling effects and large temperature variations. Therefore, we investigated the thermomechanical modeling of composite materials which takes care of mechanical loads, thermal loads, corresponding coupling effects between these for both small and large temperature variation cases and developed suitable micromechanics models for capturing these effects.

1.2 A Review of Previous Work on Micromechanics

1.2.1 Bounding Principles

The bounding of the overall moduli of composites has drawn considerable attentions before accurate solutions of effective moduli can be obtained in the field of micromechanics. The early formulation of bounds known as Voigt [1] bound which assumed the strain is uniform throughout the composites and Reuss [2] bound which assumed the stress is uniform throughout the composites. Later on, based on contribution of these two works, Hill [3] summarized the elastic behavior of composites should be within the domain between Reuss and Voigt values. Then by applying the variational principles in the linear theory of elasticity, Hashin and Shtrikman [4, 5] derived upper and lower bounds of effective moduli of multiphase materials of arbitrary phase geometry. Following that, Willis [6] further developed generalized Hashin-Shtrikman bounds and compared with self-consistent estimates for anisotropic composites. After that, bounds with three and more correlation functions (aka higher order bounds) are generated independently by applying different perturbation expressions of stress and strain field, such as Beran and Molyneux [7], McCoy [8], Silnutzer [9], Milton [10–13], Berrymen [14–16], etc. The detailed explanation of these bounds will be addressed in the following sections specifically.

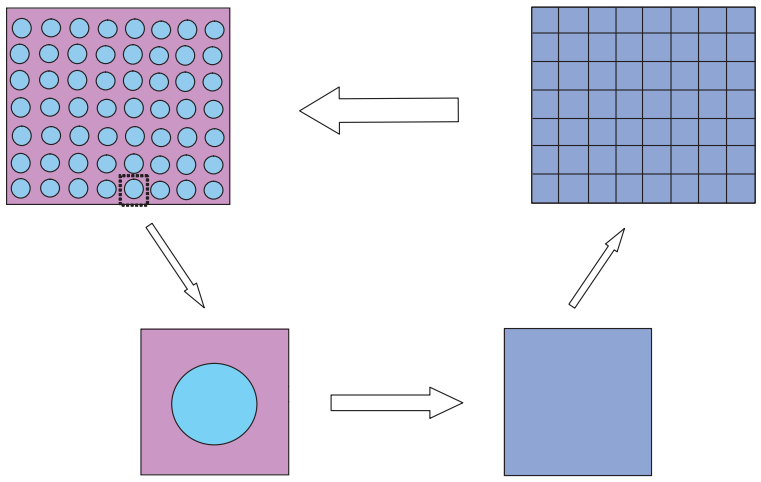

Fig. 1.1: A three step diagrammatic sketch of homogenization

Voigt and Reuss Bounds

The Voigt upper bound and Reuss lower bound are properly the earliest remarkable bounds of overall moduli of composites which apply the assumption of uniform strain and stress within unit cell. They are also known as the rules of mixture approaches to obtain effective properties of composites. The overall moduli of composites are only considered to have a one-point correlation functions which means only the volume fraction is taking into account in this case. According to Voigt and Reuss's assumptions respectively, it is easy to obtain the expressions of effective moduli of composites by applying the generalized Hookie's law and its reverse form.

For isotropic composites, according to Voigt's rule of mixture, the overall bulk moduli ($\bar{K}_V$) and shear moduli ($\bar{G}_V$) can be written as mixtures of constituents bulk and shear moduli in terms of corresponding volume fraction of the fiber (v_f) as:

$$\begin{aligned} \bar{K}_V &= v_f K_f + (1 - v_f) K_m \\ \bar{G}_V &= v_f G_f + (1 - v_f) G_m \end{aligned} \tag{1.1}$$

While according to Reuss's rule of mixture, it is easy to obtain the following relations by applying the strain-stress relation:

$$\frac{1}{\bar{K}_R} = \frac{v_f}{K_f} + \frac{1 - v_f}{K_m} \tag{1.2}$$

$$\frac{1}{\bar{G}_R} = \frac{v_f}{G_f} + \frac{1 - v_f}{G_m} \tag{1.3}$$

from Eqs. (1.2) and (1.3), the overall bulk moduli ($\bar{K}_R$) and shear moduli ($\bar{G}_R$) can be obtained as:

$$\begin{aligned} \bar{K}_R &= \frac{K_f K_m}{v_f K_m + (1 - v_f) K_f} \\ \bar{G}_R &= \frac{G_f G_m}{v_f G_m + (1 - v_f) G_f} \end{aligned} \tag{1.4}$$

Hill [3] proved that the Voigt moduli are greater than Reuss moduli and the true values of effective properties of composites ($\bar{K}_{eff}$) and ($\bar{G}_{eff}$) should lie between those two sets of bounds.

Hashin-Shtrikman Variational Bounds

Taking advantage of the variational principles which involve the polarization field, specifically, energy minimization principles, Hashin and Shtrikman derived bounds on effective material properties. These bounds are commonly considered to be the best bounds when only volume fraction is provided as the geometric information. For a two-phase composites, if we assume constituent material properties $K_2 > K_1$ and $G_2 > G_1$, the corresponding

Hashin-Shtrikman lower bounds ($\bar{K}_L$, $\bar{G}_L$) and upper bounds ($\bar{K}_U$, $\bar{G}_U$) can be given by:

$$\begin{aligned}
\bar{K}_L &= K_1 + \frac{v_2}{\frac{1}{K_2-K_1} + \frac{3v_1}{3K_1+4G_1}} \\
\bar{G}_L &= G_1 + \frac{v_2}{\frac{1}{G_2-G_1} + \frac{6(K_1+2G_1)v_1}{5G_1(3K_1+4G_1)}} \\
\bar{K}_U &= K_2 + \frac{v_1}{\frac{1}{K_1-K_2} + \frac{3v_2}{3K_2+4G_2}} \\
\bar{G}_U &= G_2 + \frac{v_1}{\frac{1}{G_1-G_2} + \frac{6(K_2+2G_2)v_2}{5G_2(3K_2+4G_2)}}
\end{aligned} \tag{1.5}$$

where v_1 and v_2 are the volume fraction of the two phases.

Improved Bounds with Two and More-Points Correlation Functions

In order to seek more accurate solutions of effective properties of multi-phase media, an improved set of bounds is required which means the gap between upper and lower bounds needs to be reduced. Improved bounds are defined as bounds depending nontrivially upon two points and higher order correlation functions by Torquato [17] which commonly include those bounds more stringent than Hashin-Shtrikman variational bounds. As more and more researchers enrolled in the field of studying improved bounds, many of them figured that knowing more geometric information of heterogeneous medium beyond volume fraction will narrow the bounds of the effective material properties as given by Hashin and Shtrikman. Beran and Molyneux [7], and McCoy [8] used perturbation expansions that have been modified by the inclusion of multiplicative constants as trial functions of stress and strain fields and generated bounds with three-points correlation functions independently.

One disadvantage of these bounds with three-points correlation functions mentioned above is that upper and lower bounds depend on different correlation functions. So Milton [13] further simplified these bounds according to the similar work done by Miller [18].

The simplified bounds on the bulk modulous ($\bar{K}_{eff}$) and shear modulous ($\bar{G}_{eff}$) of a two-phase composites can be expressed as:

$$\begin{aligned}
\bar{K}_L &= \left[\left\langle \frac{1}{K} \right\rangle - \frac{4v_1v_2(\frac{1}{K_1}-\frac{1}{K_2})^2}{4\left\langle \frac{1}{\widetilde{K}} \right\rangle + 3\left\langle \frac{1}{G} \right\rangle_\zeta}\right]^{-1} \\
\bar{G}_L &= \left[\left\langle \frac{1}{G} \right\rangle - \frac{v_1v_2(\frac{1}{G_1}-\frac{1}{G_2})^2}{\left\langle \frac{1}{\widetilde{G}} \right\rangle + 6\Xi}\right]^{-1} \\
\bar{K}_U &= \left[\langle K \rangle - \frac{3v_1v_2(K_1-K_2)^2}{3\left\langle \widetilde{K} \right\rangle + 4\langle G \rangle_\zeta}\right] \\
\bar{G}_U &= \left[\langle G \rangle - \frac{6v_1v_2(G_1-G_2)^2}{6\left\langle \widetilde{G} \right\rangle + \Theta}\right]
\end{aligned} \tag{1.6}$$

where $v_1 = 1 - v_2$ is the volume fraction of constituent material 1, and we define:

$$\begin{aligned}
\Xi &= \frac{\left[10\langle K \rangle^2 \left\langle \frac{1}{K} \right\rangle_\zeta + 5\langle G \rangle \langle 3G+2K \rangle \left\langle \frac{1}{G} \right\rangle_\zeta + \langle 3K+G \rangle^2 \left\langle \frac{1}{G} \right\rangle_\eta\right]}{\langle 9K+8G \rangle^2} \\
\Theta &= \frac{\left[10\langle G \rangle^2 \langle K \rangle_\zeta + 5\langle G \rangle \langle 3G+2K \rangle \langle G \rangle_\zeta + \langle 3K+G \rangle^2 \langle G \rangle_\eta\right]}{\langle K+2G \rangle^2}
\end{aligned} \tag{1.7}$$

and the corresponding angle brackets are defined as:

$$\begin{aligned}
\langle A \rangle &= A_1v_1 + A_2v_2 \\
\langle A \rangle_\zeta &= A_1\zeta_1 + A_2\zeta_2 \\
\langle A \rangle_\eta &= A_1\eta_1 + A_2\eta_2 \\
\left\langle \widetilde{A} \right\rangle &= A_2v_1 + A_1v_2
\end{aligned} \tag{1.8}$$

with $\zeta_1 = 1 - \zeta_2$ and $\eta_1 = 1 - \eta_2$

Later on, Milton and Phan-Thien [19] derived bounds with four-points correlation functions which are based on the Fourier series in order to achieve a considerable mathematical

simplification. The Milton and Phan-Thien bound ($\bar{K}_{mp}$) can be expressed as:

$$\bar{K}_{mp} = \left[\langle K \rangle - \frac{3v_1v_2(K_1 - K_2)^2}{3\left\langle \widetilde{K} \right\rangle + 4\Phi} \right] \tag{1.9}$$

where

$$\begin{aligned} \Phi =& \frac{G_2(\langle G \rangle_\zeta + \zeta_2(G_1 - G_2)H_2)}{G_2 + \zeta_2(G_1 - G_2)H_2} \\ H_2 =& B^* - \frac{(3K_2 + G_2)C^*}{3K_2 + 4G_2} \\ B^* =& \frac{3(B - \frac{A^2}{v_1v_2})}{\zeta_1\zeta_2v_1v_2} \\ C^* =& \frac{3(C - \frac{A^2}{v_1v_2})}{\zeta_1\zeta_2v_1v_2} \end{aligned} \tag{1.10}$$

with $\zeta_1 = 1 - \zeta_2$. The parameters (A, B, and C) are geometrical parameters defined by Fourier series and they are listed and discussed in [19] which will not be repeated in this dissertation. Eq. (1.9) represents a lower bound for bulk modulous ($\bar{K}_{eff}$) if $K_1 > K_2$ and $G_1 > G_2$ while the upper bound is obtained by interchanging all subscripts 1 and 2 in Eqs. (1.9) and (1.10).

Bounds on effective shear moduli are more complicated compared with effective bulk moduli shown above. In a simplified form, the Milton and Phan-Thien bound ($\bar{G}_{mp}$) can be expressed as:

$$\bar{G}_{mp} = \left[\langle G \rangle - \frac{6v_1v_2(G_1 - G_2)^2}{3\left\langle \widetilde{K} \right\rangle + \Phi_2} \right] \tag{1.11}$$

where

$$\Phi_2 = \frac{(9K_2 + 8G_2)(3K_1 + 4G_2) + 12G_2(K_1 - K_2)\zeta_1}{(K_2 + 2G_2)(3K_1 + 4G_2) - 2G_2(K_1 - K_2)\zeta_1} \tag{1.12}$$

where using the same method as mentioned for $\bar{K}_{eff}$, the upper and lower bounds of $\bar{G}_{eff}$ can be obtained. Milton and Phan-Thien [19] showed that these fourth order bounds are tighter than all of the second and third order bounds especially that the upper bound is dramatically lower than others. It is also worth to mention that Torquato [17,20–22] proposed the concept of n-points correlation functions by utilizing a n-point probability functions of

finding subset of n-points in the matrix phase and brought up the idea of obtaining bounds with n-points correlation functions. However, after years of study, Torquato also pointed out the corresponding difficulties of seeking these higher order bounds.

1.2.2 Homogenization of the Thermoelastic Problem

The relationships between the overall macroscopic material properties of composites and properties of individual constituent materials had been studied not only for mechanical problems but also for many thermal problems in the literature. The most classical and mature theory in this field is the theory of thermoelasticity which is first attributed to the work of Duhamel [23] in 1838 by introducing the temperature gradients in the strain expression of an elastic body. Traditional thermoelasticity theory is guided by the first and second laws of thermodynamics which can be found in many continuum mechanics textbook such as [24, 25]. Thomson [26] in 1857 used these two laws of thermodynamics to demonstrate that thermal strain and stress will be generated if temperature of an elastic body is changed. The first law of thermodynamics refers to a thermal energy balance statement that the rate of work of the internal forces equals to the increasing internal energy minus the external absorbed heat while the second law of thermodynamics concerned entropy that can be expressed in many specific ways. The most famous two statements are the Kelvin-Planck statement and the Clausius statement. In the Kelvin-Planck statement, a device can not be constructed to operate in a cycle and produce no other effect besides mechanical work through the exchange of heat [27]. Alternatively, in the Clausius statement, it is impossible to construct a device operating in a cycle and producing no effect other than the heat transfer [24]. In reality, unless the systems are in isothermal or adiabatic condition, real thermodynamic problems are usually linked to irreversible thermal conduction. The thermodynamics of irreversible processes has been investigated by many researchers during the past [28–32]. In the theory of irreversible thermodynamical process, the entropy flow is used to form a dissipation function to represent the irreversible properties of the medium. Also it shows that the thermoelastic potential energy can be expressed as a sum of the mechanical potential energy from classical elasticity theory and the thermal induced potential

energy from thermal conduction theory.

The modern theory of thermoelasticity is derived based on the theory of irreversible thermodynamical process which includes the effect of heat transfer in a body, and stress and strain caused by temperature gradients. Moreover, the interaction between thermal and mechanical effects in a body is taken into account. However, whether the coupled heat equation can be linearized to obtain fully linear set of equations of motion was still concerned at that time. Speziale [33] showed that the energy equation in the form of coupled heat equation can be linearized even if the temperature variation is large. This investigation not only solved proposed linearization problem but also relaxed the assumption made in conventional thermoelasticity that the temperature variation needs to be small. Kovalenko [34] also showed it is possible to derive theory of linear thermoelasticity by only introducing a small overall strain field without assuming the smallness of the increase in temperature respect to the starting temperature. The theoretical formulation of Kovanlenko's small-strain thermoelasticity will be shown in details in Chapter 2. However, most of these approaches are based on use of temperature independent constituents, and temperature dependent cases are less investigated. It is shown by Nadeau [35] that the expression of effective thermal expansions of composites with temperature-dependent constituents is not the same as the temperature independent ones. In order to further investigate effective thermal properties of composites, we are looking into the homogenization technique in the field of thermoelasticity in literature.

Although some of the effective properties can be determined and tested directly by experiments, there are cases especially for composites with large dimension size in one or more directions that some of the effective properties are not easy to obtain using direct testing or measurement. In this situation, the homogenization of thermoelastic problem has also been studied for prediction of effective thermomechancial properties of various composite structures since early 1960s. Different models are proposed during that period and typical reviews can be found in literature [36–40]. In general, there are plenty of different methods of homogenization available in literature but here we can classify them into

seven categories in order to summarize them: (1) mechanics of materials method; (2) fiber substructuring method; (3) self-consistent method; (4) Mori-Tanaka method; (5) method of cells; (6) other methods and apporaches; and (7) variational asymptotic method which will be used to guide all the work through this dissertation. In the following paragraphs, each method is introduced where the basic assumptions and principles of the corresponding method are reviewed and discussed respectively.

Mechanics of Materials Method

The micromechanics model presented using mechanics of material method had been presented since middle of twentieth century. This method is based on equations derived from mechanics of materials formulation by applying force equilibrium and displacement compatibility. These derived sets of equations are also mentioned as the formal structure of composite micromechanics theory in literature. This branch of work refers the properties of constituent materials, geometric configuration of composites and fabrication process as inputs, and effective mechanical and thermal properties of the composites as outputs. There are three basic assumptions in this approach: (1) Both fiber and matrix are assumed to be linear elastic and the stress-strain relation obeys Hooke's law; (2) The composite materials are considered to be macroscopic homogeneous, transversely isotropic; (3) The bonding between constituent materials is perfect and no debonding effect is considered. Numerous approaches can be found in literature but we focus on work in the field of thermoelastic problem. Greszczuk [41] derived equations for effective coefficients of thermal expansion of both fiber and matrix (α_f and α_m) for a square array composite under plane stress condition but no experimental results are available to validate his effective thermal properties. Abolin'sh [42] predicted effective thermal material properties by assuming composites to be transversely isotropic in plane normal to fibers and the Poisson effects with longitudinal load to be negligibly small. Chamis [43] derived a unified set of equations for mechanical, thermal and hygral properties, and Caruso and Chamis [44] validated all the properties by comparing with three-dimensional FEA results. Following the work of Chamis, Hopkins and Chamis [45] further extended these formulations to deal with high temperature composites

by considering large local stress excursions and temperature dependent material effects. The corresponding thermal properties obtained can be expressed as:

$$\begin{aligned}
\bar{\alpha}_{11} =& \frac{v_f \alpha_{f11} E_{f11} + v_m \alpha_m E_m}{\bar{E}_{11}} \\
\bar{\alpha}_{22} =& \bar{\alpha}_{33} = \alpha_{f22}\sqrt{v_f} + (1-\sqrt{v_f})(1+v_f \nu_m E_{f11})\alpha_m \\
\bar{K}_{11} =& v_f K_{f11} + v_m K_m \\
\bar{K}_{22} =& \bar{K}_{33} = (1-\sqrt{v_f})K_m + \frac{K_m\sqrt{v_f}}{1-\sqrt{v_f}(1-\frac{K_m}{K_{f22}})} \\
\bar{C} =& \frac{1}{\bar{\rho}}(v_f \rho_f C_f + v_m \rho_m C_m)
\end{aligned} \tag{1.13}$$

where $\bar{\alpha}$ are the effective coefficients of thermal expansion, $\bar{K}$ are the effective heat conductivities, $\bar{C}$ is the effective heat capacity, $\bar{\rho}$ is the effective density, and ν are the poisson's ratios. Quantities with subscripts $f11$, $f22$, and $f33$ stand for the fiber properties in three primary directions with $f11$ is along the fiber direction and m is the matrix. It is noticed that if the composites are void free, the volume ratios obey $v_f + v_m = 1$.

Fiber Substructuring Method

The mechanics of materials method used to be considered as the standard in the field of composite micromechanics. However, the unit cells modeled using this method contain multiple fibers (usually the whole ply). To better capture the local details of the composites, the unit cell is further subdivided and described into several slices. Each slice only contains a single fiber with the matrix. The equations derived from mechanics of materials are generated for each slice separately and then all these equations are integrated to obtain the overall effective properties for the ply or the whole composites. This method is actually an improved method using mechanics of materials by considering a much smaller slice of unit cell as the smallest representative unit instead of the whole unit cell used in mechanics of materials approach. The detailed principle and assumptions can be easily found in textbook of composite materials [46, 47]. This method has several advantages compared with conventional methods: (1) capture better local stress and strain fields; (2) account for

fiber and matrix cracking. Since the fiber and matrix cracking are always considered to be the critical issues in ceramic matrix composites so this method had been widely applied in this field [48]. This approach can deal not only fiber and matrix but also the interphase between those two constituents. Apply subscript i as the quantities of interphase and keep the other symbols including the coordinate system same as those in mechanics of materials method, the effective thermal properties of fiber substructuring method can be written as:

$$\begin{aligned}
\bar{\alpha}_{11} =& \frac{v_f \alpha_{f11} E_{f11} + v_m \alpha_{m11} E_{m11} + v_i \alpha_{i11} E_{i11}}{\bar{E}_{11}} \\
\bar{\alpha}_{22} =& v_f \alpha_{f22} + v_m \alpha_{m22} + v_i \alpha_{i22} \\
\bar{\alpha}_{33} =& v_f \alpha_{f33} + v_m \alpha_{m33} + v_i \alpha_{i33} \\
\bar{K}_{11} =& v_f K_{f11} + v_m K_{m11} + v_i K_{i11} \\
\bar{K}_{22} =& \frac{K_{f22} K_{m22} K_{i22}}{v_f K_{m22} K_{i22} + v_m K_{f22} K_{i22} + v_i K_{f22} K_{m22}} \\
\bar{K}_{33} =& \frac{K_{f33} K_{m33} K_{i33}}{v_f K_{m33} K_{i33} + v_m K_{f33} K_{i33} + v_i K_{f33} K_{m33}} \\
\bar{C} =& \frac{1}{\bar{\rho}} (v_f \rho_f C_f + v_m \rho_m C_m + v_i \rho_i C_i)
\end{aligned} \tag{1.14}$$

where the effective density $\bar{\rho}$ can be obtained by applying rule of mixtures:

$$\bar{\rho} = v_f \rho_f + v_m \rho_m + v_i \rho_i \tag{1.15}$$

Self-Consistent Method

The self-consistent method is developed on the benefit of Eshelby's transformation to an auxiliary problem where a single ellipsoidal inhomogeneous inclusion is embedded in an infinite uniformly loaded medium [49]. This method is originally proposed by Hershey [50] and Kröner [51] for aggregates of crystals. Hill [52, 53] and Budiansky [54] extended the self-consistent method and derived effective material properties for multiphase composites respectively. Applying the solution of an inclusion embedded in an infinite medium, they treated the inclusion as the fiber and the medium as the matrix in order to generate the solution for effective properties of composites. Still, the bonding between fibers and matrix

is assumed to be perfect. Later on, Budiansky [55] further implemented this method to obtain thermal constants of macroscopically isotropic composite materials.

It is shown by Gubernatis [55] that self-consistent method is an implicit method which requires iteration process during calculation. The correction factor of effective properties varies with the solution of effective properties while the total scattering among all the grains caused by correction factor also determines the solution. Laws [56] derived the effective thermal properties for n-phase composites of the thermostatic problem using self-consistent method. Moreover, for viscoelastic particulate composites with extremely high volume fractions of particles and large modulus contrast between particles and binder, Banerjee and Adams [57] concluded that only the effective CTE is close to the experimental results while the bulk modulus and shear modulus are way off the trend using self-consistent method.

Mori-Tanaka method

This method was proposed by Mori and Tanaka [58] in 1973 which is an explicit method to calculate the average internal stress in the matrix of composites. Based on the idea of Eshelby [49] on the equivalent inclusion, Benveniste [59] extended the Mori-Tanaka method and derived equivalent inclusion-average stress method (EIAS) to calculate the effective properties of composites. This approach simply follows Esheby's assumption that the s-train field is homogeneous within an ellipsoidal inclusion embedded in an infinite medium subjected to homogeneous displacement or traction boundary conditions by assuming an average strain field will be given if a single inclusion in a matrix is under homogeneous displacement or traction boundary conditions. Weng [60] showed that results obtained by Mori-Tanaka method lie within Hashin-Shtrikman bounds in order to validate this method. Berryman and Berge [61] further compared this method with self-consistent method and Kuster-Toksöz method [62] which is another explicit method used in the field of geophysics. They concluded that the Mori and Tanaka method and Kuster-Toksöz method are capable of getting the effective material properties of composites when the volume fraction of host material is at least $70 - 80\%$ and both results agreed with each other pretty well.

As for effective thermal properties, Norris [63] gave the exact equation for effective

thermal conductivity for multiphase isotropic composites and proved that effective thermal conductivity results using Mori-Tanaka method always obey Hill-Hashin bounds for two phase isotropic composites. The exact equation of effective conductivity can be written as:

$$\bar{K} = K_0 + \frac{\sum_{j=1}^{n}(K_j - K_0)v_j\frac{\bar{H}_j}{\bar{H}_0}}{v_0 + \sum_{i=1}^{n} v_i\frac{\bar{H}_i}{\bar{H}_0}} \tag{1.16}$$

where subscripts $0, 1, 2, ..., n$ stand for $n+1$ phase composites and $\bar{H}$ is the spatial average of quantity H which is related with temperature field ϕ and can be expressed as:

$$\bar{H}_i = \langle H_i \rangle = \langle -\boldsymbol{\nabla}\phi_i \rangle \tag{1.17}$$

However, Norris found that this approach for a multiphase media is outside the limits of the Hashin-Shtrikman bounds and he concluded this disagreement is due to the possible wrong approximation on the ratio $\frac{\bar{H}_j}{\bar{H}_0}$. Later on, Benveniste et al. [64] derived the expressions for effective thermal conductivities of composites reinforced with coated carbon fiber but no further comparison with Hashin-Shtrikman bounds is provided. Recently, Böhm [65] used Mori-Tanaka method to generate effective thermal conductivity of composites reinforced by non-uniformly sized particles with interfacial resistances. He found that results by Mori-Tanaka method underestimates the effective conductivities for large particles and overestimates them for small ones, and with the particle diameter in the range from 10^{-7}m to 10^{-4}m, the effective conductivities by Mori-Tanaka method lie between three-point bounds by Torquato and Rintoul [66].

Method of Cells

The method of cells (MOC) is developed by Aboudi [67] based on the assumption that fibers in a unidirectional composite can be arranged regularly in the matrix to form a doubly periodic array. With this periodic distribution in medium, the representative cell is divided into four subcells. The fiber is stored in one subcell and the rest three are

the matrix. The displacement and traction fields are continuous for all interfaces between subcells. The variation of displacement field is linear in each subcell under the microscopic coordinates located at the geometric center of each subcell [68]. The detailed formulations and explanations of MOC method are later summarized and published in book [69].

The generalized method of cells (GMC) is developed later by discretizing the RVE into a number of subcells instead of only using four in MOC [70]. With exactly the same assumptions and boundary conditions, this method favors the shape of fibers and matrix to increase the flexibility on the choice of RVE. This also enables GMC method to analyze porous composites, composites with various shape of fiber and even damage inside the RVE. One advantage of GMC is that instead of calculating effective properties of the whole RVE step by step in each subcell, the full set of effective properties is obtained by using a recursive process among all the subcells in one step. Because of this recursive process, GMC method is more computational efficient than traditional FEM. The local fields predicted by GMC is not accurate for the reason of ignorance of coupling between macroscopically applied normal stresses (strains) and the resulting microscopic shear stresses. To overcome this, an improved method called high-fidelity generalized method of cells (HFGMC) had been developed [71, 72]. The shear coupling was accomplished by expanding displacement vector into quadratic form in terms of its local coordinates in each subcells instead of a linear displacement field which has been used in both MOC and GMC. However, the computational efficiency for this method is sacrificed for the improved local field accuracy. Moreover, since the subcells are only considered to be rectangular shape, it is sometimes hard for MOC, GMC, and HFGMC to capture composites with complicated geometry in microscopic level such as woven composites and foam composites.

Other Methods and Approaches

There are massive methods and approaches in the field of micromechanics invented and investigated by individual researchers in literature. They are also worth to mention as different aspects of contribution they made in the study of homogenization of composites. The dilute method (aka dilute concentration method) [73] which considers a particle with

small portion of volume fraction in an infinite medium and no interaction between adjacent particles is considered. The differential method is a kind of an "infinitesimal" implementation of the self-consistent method. But unlike self-consistent method, in a two-phase composite one phase is matrix and the other phase is added incrementally in such a way that the newly added material is always in dilute concentration with respect to the current configuration [74, 75]. Another approach is called elasticity-based cell model (ECM) which is similar as HFGMC theory but the displacement field in each subcell is given as an infinite series [76, 77]. A detailed comparison among MOC, HFGMC, ECM, and VAMUCH will be introduced in next section which can also be found in paper [78]. There are also researchers apply finite element approaches using the conventional stress analysis of RVE to obtain the effective mechanical and thermal properties of composites which we will not discuss in details here [79].

Variational Asymptotic Method

Variational asymptotic method is a newly developed method of homogenization introduced by Prof. Berdichevsky [80] in order to investigate functionals with small parameters involved. Like the conventional asymptotic method, VAM writes out asymptotic expansions of physical problems and discards all the small negligible terms. However, to be more efficient than the conventional asymptotic method, VAM carries out asymptotic analysis of the variational statement to find the stationary point of the corresponding functional. One of the advantages of VAM is that it only deals with one functional instead of a group of differential equations like conventional asymptotic method. Also the accuracy along with the efficiency has been demonstrated repeatedly for this method.

VAM is applicable to micromechanics since (1) the size of RVE is always much smaller than the macroscopic size of the composites. Along with another two hypotheses: (2) the exact solutions of field variables have volume average over RVE; (3) effective material properties are independent of loads, boundaries, and geometries in macroscopic level, a new micromechanics model called Variational Asymptotic Method for Unit Cell Homogenization (VAMUCH) has been developed by Yu [81–83] for predicting effective properties

of composites and recovering the local stress and strain fields within the RVE accordingly. The dimensionality of unit cell (UC) used in VAMUCH is decided by the periodicity of the microstructure. For example, a binary composite formed by orthotropic layers, the materials are uniform within binary layers and only periodic along thickness direction, a 1D UC is sufficient for VAMUCH to obtain the completed set of 3D effective mechanical, thermal, piezoelectric, electromagnetic properties, and corresponding local stress and strain field within the UC. Moveover, since this model applies no limitation on the geometric formulation of the microstructure, it is capable of dealing with composites with complicate microstructures such as woven composites and sandwich structures [84].

1.2.3 Present Work and Outline of the Dissertation

In the previous sections, we have observed the motivation and necessarity of enabling micromechanics modeling of thermomechanical problems with temperature dependent constituents, both internal and external loads, large temperature variation, and even a nonuniformly distributed temperature field. In this dissertation, our efforts will be putting on the improvement of VAMUCH related with these problems. We will relax the limitations of current thermomechanical model in VAMUCH step by step. The dissertation will be organized in the following way:

- Chapter 2 introduces the theoretical formulation of the present work.
- Chapter 3 describes the improved thermomechanical micromechanics model dealing with temperature dependent constituent materials and finite temperature variations.
- Chapter 4 presents the improved thermomechanical micromechanics model dealing with nonuniformly distributed loads and temperature fields.
- Chapter 5 summarizes the work done in the dissertation and gives suggestions for related future research.

Chapter 2

Theoretical Formulation

This chapter briefs all the formulations needed to guide the current study in this dissertation including the basics of thermoelasticity theory and micromechanics. The formulations will be shown by following the order of fundamental to advanced in order to demonstrate the theories in a clear way. The formulations are reviewed and summarized by author based on understanding and further interpretations of such theories need to go to corresponding reference as referred below.

2.1 Basics of Thermoelasticity Theory

The development of thermoelasticity theory is closely related with the thermodynamics laws especially the first and second laws of thermodynamics. However, a problem of irreversible processes of thermal conduction has been set up since 1950s. It has been discussed by Boley and Weiner [85] on the structure of the constitutive equations and they found out that the equations of classical thermodynamics remain valid in the thermodynamics of irreversible processes for a local thermodynamic equilibrium. A Fourier relation between heat flux and temperature gradient can describe this irreversible thermal conduction which enabled the derivation of theory of thermoelasticity. To help understanding this theory, we introduce the laws of thermodynamics first in the following sections.

2.1.1 The First Law of Thermodynamics

The first law of thermodynamics is also commonly known as the law of conservation of energy in a thermodynamic system. The idea of law of conservation of energy is that in an isolated system, the total amount of energy can not be changed (increase or decrease), however those energies can be changed into various forms (potential energy, kinetic energy

and etc). This law is arrived by following the other general laws of physics such as the law of conservation of mass, the law of conservation of linear momentum (Newton's second law of motion), and the law of conservation of angular momentum. The ideas of these similar laws are simple and based on the sense of physics but we will only focus on the derivation of the law of conservation of energy here.

In the study of mechanics of rigid bodies, we know the kinetic energy and potential energy can be fully transformed from one to the other if energy is assumed to not dissipate. When this applies to a thermodynamics system, it is the first law of thermodynamics, which is usually stated as the time rate of change of the total energy of an isolated system is equal to the sum of heat content per unit time supplied to the system and the rate of work done by external forces acting on the system. In short words, the total energy (including kinetic energy and internal energy) change is the sum of the external work and heat changes to the system. Using the above definition, we can express the first law of thermodynamics as:

$$\frac{\mathrm{d}}{\mathrm{d}t}(K+U) = W + H \tag{2.1}$$

Now let's look at the energy terms in Eq. (2.1) respectively. The kinetic energy (K) of the system is given by:

$$\begin{aligned} K &= \frac{1}{2}\int_{\Omega} \rho \mathbf{v}\cdot\mathbf{v}\mathrm{d}\Omega \\ &= \frac{1}{2}\int_{\Omega} \rho v_i v_i \mathrm{d}\Omega \end{aligned} \tag{2.2}$$

where ρ is the density per unit volume, Ω is the volume of the continuum, $\mathbf{v}$ is the velocity field, and v_i are the components of $\mathbf{v}$. And if we set e as the energy per unit mass, the total internal energy (U) of the continuum is given by:

$$U = \int_{\Omega} \rho e \mathrm{d}\Omega \tag{2.3}$$

It is noted that the elastic strain energy or other forms of energy should be claimed as parts of internal energy (U). The rate of work done by external forces (W) consists of two parts: the rate of work done by surface traction ($\mathbf{t}$) and the rate of work done by body

force per unit mass ($\mathbf{b}$). It can be expressed as:

$$\begin{aligned} W &= \int_\Gamma \mathbf{t}\cdot\mathbf{v}\mathrm{d}\Gamma + \int_\Omega \rho\mathbf{b}\cdot\mathbf{v}\mathrm{d}\Omega \\ &= \int_\Gamma t_i v_i \mathrm{d}\Gamma + \int_\Omega \rho b_i v_i \mathrm{d}\Omega \\ &= \int_\Omega \Big[(\sigma_{ji}v_i)_{,j} + \rho b_i v_i\Big]\mathrm{d}\Omega \\ &= \int_\Omega (\sigma_{ji,j}v_i + \sigma_{ji}v_{i,j} + \rho b_i v_i)\mathrm{d}\Omega \end{aligned} \tag{2.4}$$

where divergence theorem is applied to the first term which is related with surface traction in Eq. (2.4) and Γ denotes the corresponding surface. Let $\mathbf{q}$ be the heat flux vector and r be the internal heat generation per unit mass per unit time, then the heat content per unit time (H) can be written as:

$$\begin{aligned} H &= \int_\Omega \rho r \mathrm{d}\Omega - \int_\Gamma \mathbf{q}\cdot\mathbf{n}\mathrm{d}\Gamma \\ &= \int_\Omega \rho r \mathrm{d}\Omega - \int_\Gamma q_i n_i \mathrm{d}\Gamma \\ &= \int_\Omega (\rho r - q_{i,i})\mathrm{d}\Omega \end{aligned} \tag{2.5}$$

Substituting Eqs. (2.2), (2.3), (2.4), and (2.5) into Eq. (2.1), we have:

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega (\frac{1}{2}\rho v_i v_i + \rho e)\mathrm{d}\Omega &= \int_\Omega (\sigma_{ji,j}v_i + \sigma_{ji}v_{i,j} + \rho b_i v_i + \rho r - q_{i,i})\mathrm{d}\Omega \\ \int_\Omega (\rho\frac{\mathrm{d}v_i}{\mathrm{d}t}v_i + \rho\frac{\mathrm{d}e}{\mathrm{d}t})\mathrm{d}\Omega &= \int_\Omega (\sigma_{ji,j}v_i + \sigma_{ji}v_{i,j} + \rho b_i v_i + \rho r - q_{i,i})\mathrm{d}\Omega \end{aligned} \tag{2.6}$$

From the law of conservation of linear momentum (Newton's second law of motion), we know the time rate of change of the total momentum of the body equals the vector sum of all the external forces acting on the body, that is:

$$\begin{aligned} \int_\Gamma \mathbf{t}\mathrm{d}\Gamma + \int_\Omega \rho\mathbf{b}\mathrm{d}\Omega &= \frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega \rho\mathbf{v}\mathrm{d}\Omega \\ \int_\Gamma t_i\mathrm{d}\Gamma + \int_\Omega \rho b_i\mathrm{d}\Omega &= \int_\Omega \rho\frac{\mathrm{d}v_i}{\mathrm{d}t}\mathrm{d}\Omega \\ \int_\Gamma (\sigma_{ji,j} + \rho b_i - \rho\frac{\mathrm{d}v_i}{\mathrm{d}t})\mathrm{d}\Omega &= 0 \end{aligned} \tag{2.7}$$

where the integrand must vanish for the integral to hold for arbitrary body Ω, so we have:

$$\sigma_{ji,j} + \rho b_i = \rho \frac{\mathrm{d}v_i}{\mathrm{d}t} \tag{2.8}$$

Substituting Eq. (2.8) into Eq. (2.6), we have:

$$\int_\Omega (\sigma_{ji,j} v_i + \rho b_i v_i + \rho \frac{\mathrm{d}e}{\mathrm{d}t}) \mathrm{d}\Omega = \int_\Omega (\sigma_{ji,j} v_i + \sigma_{ji} v_{i,j} + \rho b_i v_i + \rho r - q_{i,i}) \mathrm{d}\Omega \tag{2.9}$$

where cancel the same terms on both sides of Eq. (2.9) and rearrange it, we can easily get:

$$\int_\Omega (\rho \frac{\mathrm{d}e}{\mathrm{d}t} - \sigma_{ji} v_{i,j} - \rho r + q_{i,i}) \mathrm{d}\Omega = 0 \tag{2.10}$$

Again since the integration domain can be arbitrary, the integrand must satisfy the following field equation:

$$\rho \frac{\mathrm{d}e}{\mathrm{d}t} - \sigma_{ji} v_{i,j} - \rho r + q_{i,i} = 0 \tag{2.11}$$

Exploiting the symmetry of the stress tensor σ_{ij}, we can show:

$$\sigma_{ji} v_{i,j} = \frac{1}{2}(\sigma_{ji} v_{i,j} + \sigma_{ij} v_{j,i}) = \sigma_{ij} [\frac{1}{2}(v_{i,j} + v_{j,i})] = \sigma_{ij} D_{ij} \tag{2.12}$$

where D is the rate of deformation tensor, then Eq. (2.11) can be expressed as:

$$\rho \frac{\mathrm{d}e}{\mathrm{d}t} - \sigma_{ij} D_{ij} + q_{i,i} - \rho r = 0 \quad \text{or} \quad \rho \frac{\mathrm{d}e}{\mathrm{d}t} - \boldsymbol{\sigma} : \mathbf{D} + \boldsymbol{\nabla} \cdot \mathbf{q} - \rho r = 0 \tag{2.13}$$

This field equation is called the local form of the energy equation (aka thermodynamic form of the energy equation), which is a direct consequence of the law of conservation of energy applied to a continuum.

2.1.2 The Second Law of Thermodynamics

The first law of thermodynamics shows us that if heat exchange happens between two bodies, the heat flow out of one body must be equal to the heat flow into the other body.

But the first law of thermodynamics does not specify the direction of the heat transfer while in fact heat will always transfer from the body with higher temperature to the body with lower temperature. And this process is irreversible without additional work. This fact is guided by the second law of thermodynamics which is related with temperature and entropy. Entropy is defined as a thermal property that measures the system's thermal energy per unit temperature and it is a function of strain and temperature such that an entropy equation of state exists [86]:

$$\mathrm{d}\eta = \left[\frac{\rho r - \boldsymbol{\nabla}\cdot\mathbf{q}}{\rho T}\right]\mathrm{d}t \tag{2.14}$$

where η is the entropy per unit mass and T is the temperature. The heat energy due to entropy and temperature is normally expressed as $-\eta T$. And we introduce the entropy production (S):

$$S = \int_\Omega \rho\eta \mathrm{d}\Omega \tag{2.15}$$

Then we can obtain the entropy input rate as:

$$\int_\Omega \frac{\rho r}{T}\mathrm{d}\Omega - \int_\Gamma \frac{q_i n_i}{T}\mathrm{d}\Gamma \tag{2.16}$$

The second law of thermodynamics puts restriction that the rate of entropy increase must be greater than the entropy input rate, which is:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega \rho\eta \mathrm{d}\Omega \geqslant \int_\Omega \frac{\rho r}{T}\mathrm{d}\Omega - \int_\Gamma \frac{q_i n_i}{T}\mathrm{d}\Gamma \tag{2.17}$$

Eq. (2.17) is the integral form of the Clausis-Duhem inequality in terms of the specific entropy [87]. Again, with the application of divergence theorem, we obtain:

$$\int_\Omega \left[\rho\frac{\mathrm{d}\eta}{\mathrm{d}t} - \frac{\rho r}{T} + \left(\frac{q_i}{T}\right)_{,i}\right]\mathrm{d}\Omega \geqslant 0 \tag{2.18}$$

Finally, the local form of the second law of thermodynamics (aka Clasius-Duhem inequality) is obtained as:

$$\frac{\mathrm{d}\eta}{\mathrm{d}t} - \frac{r}{T} + \frac{1}{\rho}\left(\frac{q_i}{T}\right)_{,i} \geqslant 0 \quad \text{or} \quad \frac{\mathrm{d}\eta}{\mathrm{d}t} - \frac{r}{T} + \frac{1}{\rho}\boldsymbol{\nabla}\cdot\left(\frac{\mathbf{q}}{T}\right) \geqslant 0 \tag{2.19}$$

2.1.3 Thermodynamic Potentials

In thermodynamics, there are four so called thermodynamic potentials that are used to describe the thermodynamic state of a system. They are internal energy, enthalpy, Helmholtz free energy, and Gibbs free energy. Internal energy is the total energy needed to create the system but excludes the energy to displace the system's surroundings; Enthalpy is a measure of the total energy including internal energy and the energy required to "make room for it" (function of pressure and volume); Helmholtz free energy measures the total energy including internal energy and the heat energy due to entropy at a constant temperature and volume; Gibbs free energy is the maximum amount of total energy obtained from a thermodynamic system at a constant temperature and pressure.

To find out the corresponding thermodynamics state functions for the system which do not invoke the surroundings, certain specific conditions such as constant temperature and volume or constant temperature and pressure are applied in the system. In the following, we will introduce the Helmholtz free energy and Gibbs Free energy which are two most important potentials in thermodynamics and of course guide the study through this dissertation.

Helmholtz Free Energy

Let us assume we have a process in a constant volume in which case the heat exchanged with the surroundings is equal to internal energy change as:

$$\mathrm{d}U = \boldsymbol{\nabla}\cdot\mathbf{q} \tag{2.20}$$

and from the Clasius-Duhem inequality, we have:

$$\mathrm{d}S \geqslant \frac{\boldsymbol{\nabla} \cdot \mathbf{q}}{T} \tag{2.21}$$

Therefore, since the absolute temperature T always greater than 0, we can rearrange this equation as:

$$\boldsymbol{\nabla} \cdot \mathbf{q} - T\mathrm{d}S \leqslant 0 \quad \text{or} \quad \mathrm{d}U - T\mathrm{d}S \leqslant 0 \tag{2.22}$$

If we further assume the temperature change $\mathrm{d}T = 0$, then we can rewrite Eq. (2.22) as:

$$\mathrm{d}(U - TS) \leqslant 0 \tag{2.23}$$

Then we define a new function F called Helmholtz free energy, such that:

$$F = U - TS \tag{2.24}$$

and $\mathrm{d}F \leqslant 0$. It is noted that a process at constant volume and temperature will reach equilibrium if $\mathrm{d}F = 0$

Gibbs Free Energy

Now let us assume we have a process happened at constant pressure. In this case, the heat exchanged with the surroundings is equal to the enthalpy. So it is necessary for us to bring up the definition of enthalpy first which is another thermodynamic potential. The enthalpy of a homogeneous system is defined as [88]:

$$H = U + p\Omega \tag{2.25}$$

where H is the enthalpy and p is the pressure of the system. Then the heat exchanged with the surroundings can be expressed as:

$$\boldsymbol{\nabla} \cdot \mathbf{q} = \mathrm{d}H \tag{2.26}$$

Make use of Eq. (2.21) and knowing the absolute temperature T always greater than 0, then at a constant pressure we obtain:

$$\mathrm{d}H - T\mathrm{d}S \leqslant 0 \tag{2.27}$$

If we further assume the temperature change $\mathrm{d}T = 0$, we can rewrite Eq. (2.27) as:

$$\mathrm{d}(H - TS) \leqslant 0 \tag{2.28}$$

Then we define a new function G called Gibbs free energy, such that:

$$G = H - TS \tag{2.29}$$

It is obvious to see that the Gibbs free energy are derived at a constant pressure and temperature. Again, if $\mathrm{d}G = 0$, a process at constant pressure and temperature will reach its equilibrium.

2.1.4 Small-Strain Thermoelasticity

It is well known that the traditional theory of thermoelasticity has restrictions on both strains and temperature variations, and considers the coupling effect between thermal and mechanical effects of a continuous body. However, after the linearization of coupled heat equation, the coupling effect becomes negligible and the relaxation time (the time needed to reach the steady-state heat conduction) increases with the temperature variation being small. This issue is of course not wanted which was discussed by Lord and Shulman [89] by conducting a lot of experiments and trying to find a reasonable temperature range. Later on, Speziale [33] found out even with a large temperature variation, the coupled heat equation can still be linearized without violating the assumption of small strains. This is because even with a large temperature variation in the system, the pure thermal strain remains below the level assumed in the linear theory of elasticity.

Kovalenko first discussed the case that he abandoned the restriction on temperature

variation and introduced a small-strain thermoelasticity theory with large temperature change in his book [90]. After that, based on this theory, Lubarda [91] and Boussaa [92] derived the expressions for thermodynamic potentials. In Kovalenko's small-strain thermoelasticity theory for isotropic solids, the Helmholtz free energy is expressed using a quadratic representation, such that:

$$F(I_1, I_2, T) = F(0,0,T) + \frac{\partial F(0,0,T)}{\partial I_1} I_1 + \frac{\partial F(0,0,T)}{\partial I_2} I_2 + \frac{1}{2}\frac{\partial^2 F(0,0,T)}{\partial I_1^2} I_1^2 \tag{2.30}$$

where I_1 and I_2 are related with strain and defined as:

$$\begin{aligned} I_1 =& \varepsilon_{kk} \\ I_2 =& \varepsilon_{ij}\varepsilon_{ij} \end{aligned} \tag{2.31}$$

And from Eq. (2.24) if we treat strain as constant, the entropy can be expressed as $S = -\frac{\partial F}{\partial T}$, so we have:

$$S(I_1, I_2, T) = -\frac{\partial F(0,0,T)}{\partial T} - \frac{\partial^2 F(0,0,T)}{\partial I_1 \partial T} I_1 - \frac{\partial^2 F(0,0,T)}{\partial I_2 \partial T} I_2 - \frac{1}{2}\frac{\partial^3 F(0,0,T)}{\partial I_1^2 \partial T} I_1^2 \tag{2.32}$$

and use the definition of stress tensor $\sigma_{ij} = \frac{\partial F}{\partial \varepsilon_{ij}}$, which is:

$$\sigma_{ij} = \frac{\partial F(0,0,T)}{\partial I_1}\delta_{ij} + 2\frac{\partial F(0,0,T)}{\partial I_2}\varepsilon_{ij} + \frac{\partial^2 F(0,0,T)}{\partial I_1^2}\varepsilon_{kk}\delta_{ij} \tag{2.33}$$

The strain free heat capacity per unit volume, $C_{\varepsilon=0}$ is defined as:

$$C_{\varepsilon=0} = T\left(\frac{\partial S}{\partial T}\right)_{\varepsilon=0} = -T\frac{\partial^2 F(0,0,T)}{\partial T^2}, \tag{2.34}$$

with

$$F(0,0,T) = \int_{T_0}^{T}\int_{T_0}^{\zeta} \frac{C_{\varepsilon=0}}{\varphi} \mathrm{d}\varphi \mathrm{d}\zeta \tag{2.35}$$

The first and second lame constants are defined as:

$$\begin{aligned}\lambda =& \frac{\partial^2 F(0,0,T)}{\partial I_1^2} \\ \mu =& \frac{\partial F(0,0,T)}{\partial I_2}\end{aligned} \tag{2.36}$$

Substituting equations in (2.36) into Eq. (2.33) and have $i = j = k$, then:

$$\varepsilon_{kk} = \frac{1}{3\lambda + 2\mu}\left[\sigma_{kk} - 3\frac{\partial F(0,0,T)}{\partial I_1}\right] \tag{2.37}$$

If $\sigma_{kk} = 0$, the strain in Eq. (2.37) will purely due to thermal expansion, and from the definition of coefficients of thermal expansion (α_T), we have:

$$3\int_{T_0}^{T} \alpha_T \mathrm{d}T = -\frac{3}{3\lambda + 2\mu}\frac{\partial F(0,0,T)}{\partial I_1} \tag{2.38}$$

where Eq. (2.38) can be rearranged and written as:

$$\frac{\partial F(0,0,T)}{\partial I_1} = -(3\lambda + 2\mu)\breve{\alpha_T}(T - T_0), \tag{2.39}$$

with

$$\breve{\alpha_T} = \frac{1}{T - T_0}\int_{T_0}^{T} \alpha_T \mathrm{d}T \tag{2.40}$$

where $\breve{\alpha_T}$ is also called the secant coefficient of thermal expansion.

Substituting Eqs. (2.35), (2.36), (2.37), and (2.40) into Eq. (2.30), we obtain the final expression of Helmholtz free energy for Kovalenko's small-strain thermoelasticity theory for isotropic solids as:

$$F = \frac{\lambda}{2}\varepsilon_{kk}^2 + \mu\varepsilon_{ij}\varepsilon_{ij} - (3\lambda + 2\mu)\breve{\alpha_T}(T - T_0)\varepsilon_{kk} - \int_{T_0}^{T}\int_{T_0}^{\zeta}\frac{C_{\varepsilon=0}}{\varphi}\mathrm{d}\varphi\mathrm{d}\zeta \tag{2.41}$$

As we mentioned, the derivation of Helmholtz free energy in Kovalenko's small-strain thermoelasticity theory has no restriction on the temperature change which enables this theory to deal with system with large temperature variations and materials with temperature

dependent material properties. And by further developing this theory, we derived a new micromechanics model dealing with temperature dependent material properties and large temperature variation, and the detailed derivations will be demonstrated in the Chapter 3.

2.2 Variational Asymptotic Method

We have introduced VAM in Chapter 1 in aspects of its based fundamental theories, advantages, and efficiency. In this chapter, formulations of VAM and its basic theories such as variational principles, asymptotic analysis will be reviewed.

2.2.1 Variational Principles

The variational principle is to find functions correlated with the stationary points (maximum or minimum) of functionals. The early variational principle in mechanics such as Mopertuis' variational principle is aiming at explain general phenomena of nature so it is sometimes claimed as the general law of nature. The modern variational principles where variational statements are used to decide the stationary functions for the so called integral functionals are based on the Euler's calculus of variations. The idea came from Leibniz in the eighteenth century and he found out that an action can reach not only a minimum but also a maximum value in a real process. Since the actions are often described by integral equations, the work to seek for the stationary values of this kind of functionals under integral is named as principle of least action. Nowadays, the most popular variational principle used in mechanics is the extended Hamilton's variational principle which bases on minimization of total potential energy in a dynamic system. This is called the principle of minimum potential energy (MPE) that can be derived as a special condition of principle of virtual work. This principle is introduced by Sokolnikoff [93] that the variation of potential energy in an equilibrium configuration is zero. An important outcome of principle of least action and MPE is that for a sufficiently small time increase Δt, the energy reaches minimum on the true trajectory with a finite number of degrees of freedom. A complete review of variational principles can be found in book [94].

For a function $F(y)$ in mathematics, by solving the equation of its first derivative

$F'(y) = 0$, we can either find a maximum, minimum, or saddle value of function $F(y)$ which is determined by the sign of its second derivative $F''(y)$. However, for a continuous problem, $y(x)$ is usually also a function of x. Then the solution of $F'(y(x)) = 0$ related with the stationary function $y(x)$ and the derivative of $F(y(x))$ is not that easy to find. To help understanding calculus of variations, we consider the functional $I(y)$ in the following form:

$$I(y) = \int_{x_1}^{x_2} F(x, y(x), y'(x))\mathrm{d}x \tag{2.42}$$

For a fixed value of x, if $y(x)$ is changed by a infinitesimal variations δy (aka variation of y), similarly the variation of $y'(x)$ can be denoted as $\delta y'$, then the difference of functional can be stated as:

$$\delta I = I(y + \delta y) - I(y) \tag{2.43}$$

Using Maclaurin series, we can expand the right hand side of Eq. (2.43) as:

$$\begin{aligned}\delta I = \int_{x_1}^{x_2} \Big\{ & F(x, y, y') + (\frac{\partial F}{\partial y}\delta y + \frac{\partial F}{\partial y'}\delta y') \\ & + \frac{1}{2!}\left[\frac{\partial^2 F}{\partial y^2}(\delta y)^2 + \frac{\partial^2 F}{\partial y'^2}(\delta y')^2 + \frac{\partial F}{\partial y}\frac{\partial F}{\partial y'}\delta y \delta y'\right] + ... - F(x, y, y') \Big\}\mathrm{d}x\end{aligned} \tag{2.44}$$

where the terms after second order until nth order are omitted to save space. Thus the variation of functional δI is obtained if one only keep terms of first order and drop all other higher order terms in Eq. (2.44). The variation of this functional (δI) can be rewritten as:

$$\delta I = \int_{x_1}^{x_2} (\frac{\partial F}{\partial y}\delta y + \frac{\partial F}{\partial y'}\delta y')\mathrm{d}x \tag{2.45}$$

As it is easy to notice from the assumption in the calculation of variation of functional, the only variables that can be varied are y and δy while x is fixed. To find the stationary point of the functional, we need to have $\delta I = 0$. In order to solve it, we integrate Eq. (2.45) by parts:

$$\delta I = \int_{x_1}^{x_2} \left[\frac{\partial F}{\partial y}\delta y - \frac{\mathrm{d}}{\mathrm{d}x}(\frac{\partial F}{\partial y'})\delta y\right]\mathrm{d}x + \frac{\partial F}{\partial y'}\delta y|_{x_1}^{x_2} \tag{2.46}$$

where the last term in Eq. (2.46) is related with the so called admissible constraints of the function $y(x)$ where the variations of $y(x)$ could vanish at both limits of the functional if they are prescribed at the ends:

$$\delta y(x_1) = \delta y(x_2) = 0 \tag{2.47}$$

with Eq. (2.47), it is easy to see that the last term in Eq. (2.46) goes to zero. If the left out terms of Eq. (2.46) also go to zero, we can have the desired equilibrium $\delta I = 0$ which yields:

$$\frac{\partial F}{\partial y} - \frac{\mathrm{d}}{\mathrm{d}x}(\frac{\partial F}{\partial y'}) = 0 \tag{2.48}$$

The condition expressed in Eq. (2.48) gives the stationary value of functional in Eq. (2.42) which is also known as the Euler-Lagrange equation of this problem. Taking another clear look at Eq. (2.48), if F does not depend on x explicitly, we can get:

$$\frac{\mathrm{d}}{\mathrm{d}x}(F - y'\frac{\partial F}{\partial y'}) = \frac{\partial F}{\partial y}y' + \frac{\partial F}{\partial y'}y'' - y''\frac{\partial F}{\partial y'} - y'\frac{\mathrm{d}}{\mathrm{d}x}(\frac{\partial F}{\partial y'}) \tag{2.49}$$

after cancelling the middle two terms on the right hand side of Eq. (2.49) and taking advantage of the Euler-Lagrange equation in Eq. (2.48), we obtain:

$$F - y'\frac{\partial F}{\partial y'} = const. \tag{2.50}$$

where Eq. (2.50) is called Hamiltonian. In reality, solving such Euler-Lagrange equations is not that easy as many of them do not possess analytical solutions. In order to deal with such cases, approximation methods have been developed based on variational form of the problem and the most typical one is called Rayleigh-Ritz method. The idea is to use a set of trial functions y_i to express $y(x)$ and make sure to meet the geometric boundary conditions of the problem. The function $y(x)$ can be expressed as:

$$y(x) = c_1y_1(x) + c_2y_2(x) + c_3y_3(x) + ... = \sum_{i=1}^{N} c_iy_i(x) \tag{2.51}$$

The procedure of this method is simple, first chose one set of trial functions, then treat the combination of functions as shown on the right hand side of Eq. (2.51) as $y(x)$ and substitute it into the original functional. By solving $\delta I = 0$, the values of constants $c_1, c_2, ..., c_n$ are obtained. The accuracy of solution can be improved by adding more trial functions of $y(x)$.

2.2.2 Principle of Virtual Work

Variational principles as introduced in the Section 2.2.1 are very powerful techniques for obtaining the solutions of problems in solid mechanics. Although there are many variational principles in mechanics, here we focus on introducing the Principle of Virtual Work (PVW) which will be applied to guide the study in this dissertation.

The principle of virtual work (aka principle of virtual displacement) is defined as for a system in equilibrium under the action of a number of forces (including the inertial forces), the total work done for a virtual displacement is zero. If we consider a elastic body in equilibrium under applied surface tractions and inertial body forces, imagine the elastic body will move a displacement ($\delta\mathbf{u}$). This displacement is possible but not necessarily takes place, this kind of imaginary displacement is called virtual displacement and the work done by all the forces during this virtual displacement is called virtual work. Following the definition of principle of virtual work, we can construct the following statement as:

$$-\int_{\Omega}(\sigma_{ji,j}+\rho b_i)\delta u_i \mathrm{d}\Omega + \int_{\Gamma}(n_i\sigma_{ji}-t_i)\delta u_i \mathrm{d}\Gamma = 0 \tag{2.52}$$

This statement is constructed by using the admissible stress field that both stress conditions inside parentheses are zero and considering the change of same arbitrary virtual displacement in both Γ and Ω domains. Then we use integration by parts on the first term of Eq. (2.52) as:

$$-\int_{\Omega}\sigma_{ji,j}\delta u_i \mathrm{d}\Omega = \int_{\Omega}\sigma_{ji}\delta u_{i,j}\mathrm{d}\Omega - \int_{\Gamma} n_j\sigma_{ji}\delta u_i \mathrm{d}\Gamma \tag{2.53}$$

Substituting Eq. (2.53) into Eq. (2.52) and make use of the symmetry of stress, we have:

$$\int_{\Omega} \sigma_{ji}\delta u_{i,j}\mathrm{d}\Omega = \int_{\Omega} \rho b_i \delta u_i \mathrm{d}\Omega + \int_{\Gamma} t_i \delta u_i \mathrm{d}\Gamma \tag{2.54}$$

where we can also define a virtual strain as:

$$\delta\varepsilon_{ij} = \frac{1}{2}(\delta u_{i,j} + \delta u_{j,i}) \tag{2.55}$$

Then Eq. (2.54) can be rewritten as:

$$\int_{\Omega} \sigma_{ij}\delta\varepsilon_{ij}\mathrm{d}\Omega = \int_{\Omega} \rho b_i \delta u_i \mathrm{d}\Omega + \int_{\Gamma} t_i \delta u_i \mathrm{d}\Gamma \tag{2.56}$$

where the left-hand side of Eq. (2.56) is often called internal virtual work (IVW) which stands for the work done by internal stresses while the right-hand side is called external virtual work (EVW) which stands for the work done by external applied loads.

2.2.3 Asymptotic Analysis

Instead of using Rayleigh-Ritz method to deal with the variational statement, VAM is carrying out analysis by dropping the small terms in energy expression which follows the rules of asymptotic analysis. To help identify the small terms, order symbols (aka order notations) need to be introduced. There are normally three kinds of special symbols of order: O (Big-Oh), o (little-oh), and $\sim$ (equivalent). Let $f(x)$ and $g(x)$ be real functions defined in domain $x \in [0, +\infty]$.

- We say $f(x)$ is in the order of $g(x)$ ($f = O(g)$) as $x \to 0$ if there exists certain constants C such that $|f(x)| \leqslant C|g(x)|$. In other words, function $f(x)$ is comparable in order with $g(x)$ in the neighborhood of zero point.

- We say $f(x)$ is in the small order of $g(x)$ ($f = o(g)$) as $x \to 0$ if for any small constants ε such that $|f(x)| \leqslant \varepsilon|g(x)|$. In other words, function $f(x)$ is much smaller in order compared with $g(x)$ in the neighborhood of zero point.

- We say $f(x)$ is equivalent to $g(x)$ ($f \sim o(g)$) as $x \to 0$ if $f(x) - g(x) = o[g(x)]$. This is sometimes also noted as $f(x)$ is asymptotically equal to $g(x)$.

To carry out an asymptotic analysis, the corresponding function or functional f is usually expressed as any infinite series that can decease the little-oh order of this function or functional. This kind of series are called asymptotic expansions (aka asymptotic series) of f. In order to process asymptotic analyses, these series are designated to possess only terms as good as its Big-Oh order. To identify the small little-oh terms in a functional, the asymptotic order of both the functional and its derivatives are concerned. We consider a function $f(x)$ defined as $x \in [a, b]$ and sufficiently smooth in this domain. We also denote the amplitude of change of $f(x)$ in domain as the maximum difference of the function evaluated at any two points, i.e.

$$\bar{f} = \max_{x_1, x_2 \in [a,b]} |f(x_1) - f(x_2)| \tag{2.57}$$

Then for a sufficiently small number l, the following inequality holds:

$$\left|\frac{\mathrm{d}f}{\mathrm{d}x}\right| \leqslant \frac{\bar{f}}{l} \tag{2.58}$$

The largest constant l satisfying the above inequality is called the characteristic length of function $f(x)$ in its domain. If higher derivatives need to be evaluated, a set of equations of inequalities similar as Eq. (2.58) hold:

$$\left|\frac{\mathrm{d}f}{\mathrm{d}x}\right| \leqslant \frac{\bar{f}}{l}, \quad \left|\frac{\mathrm{d}^2 f}{\mathrm{d}x^2}\right| \leqslant \frac{\bar{f}}{l^2}, \quad \cdots, \quad \left|\frac{\mathrm{d}^{\mathrm{k}} f}{\mathrm{d}x^k}\right| \leqslant \frac{\bar{f}}{l^k} \tag{2.59}$$

where k is the highest derivative of interest of the problem. And the corresponding characteristic length is the largest constant satisfying the whole set of equations of inequalities as shown in Eq. (2.59). There is also another iterative method of identifying the small parameters in an asymptotic analysis which is called the method of dominant balance. This method is simple but highly depends on the analyzers' personal experience. Instead of using the method of finding the characteristic length mentioned above, the analyzer takes a clever guess of which terms in ordinary differential equation (ODE) may be negligible and

by dropping those terms, a new function or functional is formed so as to solve the targeted ODE. After that, a consistency check needs to be performed with the guess made aforehand. If without any problems, iterative process is started by using the promising results of last iterative step as the first term of solution until the asymptotic behavior of solution is obtained.

The initial guess of the small parameters is very important as if one chooses wrong parameters, the iterative process will be unnecessarily long and complicated by doing this back and forth. But sometimes it obeys the common sense of structural mechanics. For example, in beam shaped structures, the cross section radius (r) comparing with beam length (l); in plate shaped structures, the plate thickness (t) comparing with the other two dimensions (a and b) of plate surface; in micromechanics, the microscopic dimensions of UC (y_i) comparing with the macroscopic dimensions (x_i) of the overall composites. In such cases, dropping the terms related with the small parameters will save the efforts of this initial guess.

2.2.4 Variational Asymptotic Method

For problems of physics and mechanics with small and large parameters involved, various asymptotic approaches were developed. It is clear that by using a special variational structure, there exists a direct variational approach based on asymptotic analysis of corresponding functionals which is called the variational asymptotic method. It allows us to consider minimization problems of differential equations possessing variational structure with a finite number of variables. The advantage of this method is greatly cutting down the number of equations to solve in the system to just one with a variational structure while the conventional asymptotic method deals with a complicated set of differential equations without such structure.

The basic idea of VAM is to drop the small terms in the energy expression and use variational statement to solve the stationary point of such energy functional. A systematic introduction of how to neglect small terms, how to deal with the loss of uniqueness, how those dropped small terms affect the results, and how the iteration procedure could be used

is given by Berdichevsky in his book [94]. To help understanding the basic procedure of VAM, an example as shown in book [94] is used here for explanation. Let a function $f(u,\varepsilon)$ depending on a small parameter ε be given at some set $\mathcal{C}$ of element u. Assuming that this function $f(u,\varepsilon)$ possess a stationary point denoted by $\breve{u}$ and in the form of:

$$f(u,\varepsilon) = u^2 + u^3 + 2\varepsilon u + \varepsilon u^2 + \varepsilon^2 u \tag{2.60}$$

The stationary point ($\breve{u}$) can be analytically solved as:

$$\breve{u} = \frac{1}{3}(-1-\varepsilon \pm \sqrt{1-4\varepsilon-2\varepsilon^2}) \tag{2.61}$$

where the exact solution can be expanded asymptotically in terms of ε as:

$$\breve{u} = \begin{cases} -\frac{2}{3} + \frac{\varepsilon}{3} + \varepsilon^2 + o(\varepsilon^2) \\ 0 - \varepsilon - \varepsilon^2 + o(\varepsilon^2) \end{cases} \tag{2.62}$$

Next, we use VAM to estimate the results and compare with the solution in Eq. (2.62). The zeroth order approximation using VAM is straight forward: for $\varepsilon \to 0$, the function $f(u,0) = u^2 + u^3$ has two stationary points, $u_0 = -\frac{2}{3}$ and $u_0 = 0$.

Now we need to find the first order approximation in the neighborhood of two zeroth order solutions. The procedure is set current $u = u_0 + u'$ and $u' \to 0$ for $\varepsilon \to 0$. First, we set $u = -\frac{2}{3} + u'$, the function becomes:

$$f(-\frac{2}{3} + u', \varepsilon) = -u'^2 + \frac{2u'\varepsilon}{3} + \underline{u'^3 + u'^2\varepsilon + u'\varepsilon^2} + \frac{4}{27} - \frac{8\varepsilon}{9} \tag{2.63}$$

where the last two terms are constants which are not functions of u. They will not affect the stationary points and can be simply dropped. The underlined terms are much smaller than the other two terms in view of both u and ε are tiny, and to be specific:

$$\left|u'^3\right| \ll \left|u'^2\right| \qquad \left|u'^2\varepsilon\right| \ll \left|u'^2\right| \qquad \left|u'\varepsilon^2\right| \ll \left|\frac{2u'\varepsilon}{3}\right| \tag{2.64}$$

So we drop the underlined terms and only keep two leading terms of Eq. (2.63), the function becomes:

$$f_1(u', \varepsilon) = -u'^2 + \frac{2u'\varepsilon}{3} \tag{2.65}$$

where the corresponding first order stationary point is $u' = \frac{1}{3}\varepsilon$. Note that the asymptotic order of u' is not assumed a priori, but is determined as the stationary point of the function $f_1(u', \varepsilon)$. Hence, we have obtained up to the first order approximation of the stationary point in the neighborhood of $-\frac{2}{3}$ as:

$$\breve{u} = u_0 + u_1 = -\frac{2}{3} + \frac{1}{3}\varepsilon + o(\varepsilon) \tag{2.66}$$

Then we set $u = 0 + u' = u'$ for seeking the first order approximation in the neighborhood of 0, we obtain the following function:

$$f(u', \varepsilon) = u'^2 + 2u'\varepsilon + \underline{u'^3 + u'^2\varepsilon + u'\varepsilon^2} \tag{2.67}$$

Again, the underlined terms are much smaller than those two leading terms, which is base on:

$$\left|u'^3\right| \ll \left|u'^2\right| \qquad \left|u'^2\varepsilon\right| \ll \left|u'^2\right| \qquad \left|u'\varepsilon^2\right| \ll \left|2u'\varepsilon\right| \tag{2.68}$$

in view of the fact that both u' and ϵ are small. Keep these two leading terms and drop others, we obtain the following function:

$$f_1(u', \varepsilon) = u'^2 + 2u'\varepsilon \tag{2.69}$$

where Eq. (2.69) is reaching its stationary point when $u' = -\varepsilon$. Then the solution up to the first order approximation of the stationary point in the neighborhood of 0 can be expressed as:

$$\breve{u} = u_0 + u_1 = 0 - \varepsilon + o(\varepsilon) \tag{2.70}$$

Till now, we have exactly reproduced the first two terms of asymptotic expansions of

the exact solution. We can continue this same procedure in order to find the higher order approximation. For example, to seek for the second order approximation, we similarly let $u = u_0 + u' + u''$ and $u'' \to 0$ for $\varepsilon \to 0$. First we set $u = -\frac{2}{3} + \frac{\epsilon}{3} + u''$, and similarly drop the small terms as in Eqs. (2.63) and (2.67), the following leading terms are left out in view of the fact that both u'' and ε are small:

$$f_2(u'', \varepsilon) = -u''^2 + 2u''\varepsilon^2 \tag{2.71}$$

which is reaching stationary point when $u'' = \varepsilon^2$. Then the solution up to the second order approximation of the stationary point in the neighborhood of $-\frac{2}{3}$ can be expressed as:

$$\breve{u} = u_0 + u_1 + u_2 = -\frac{2}{3} + \frac{1}{3}\varepsilon + \varepsilon^2 + o(\varepsilon^2) \tag{2.72}$$

Then we set $u = 0 - \varepsilon + u'' = -\varepsilon + u''$ for seeking the second order approximation in the neighborhood of 0, we obtain the following function by only keep the leading terms:

$$f_2(u'', \varepsilon) = u''^2 + 2u''\varepsilon^2 \tag{2.73}$$

which is reaching stationary point when $u'' = -\varepsilon^2$. Then the solution up to the second order approximation of the stationary point in the neighborhood of 0 can be expressed as:

$$\breve{u} = u_0 + u_1 + u_2 = 0 - \varepsilon - \varepsilon^2 + o(\varepsilon^2) \tag{2.74}$$

As we can see, up to the second order approximation of solution, they match the asymptotic expansions of exact solution as shown in Eq. (2.62), so an exact asymptotic expansion has been formed in this case.

Obviously, the main difficulty in VAM is to recognize the leading terms and the negligible terms same as the conventional asymptotic analysis. It is relatively easy to identify such terms in this example but in realty, to determine these relations, we need to consider the following two conditions:

- For two terms $A(u,\varepsilon)$ and $B(u,\varepsilon)$ in the functional $I(u,\varepsilon)$, if

$$\lim_{\varepsilon\to 0}\max_{u\in M}\left|\frac{B(u,\varepsilon)}{A(u,\varepsilon)}\right|=0 \tag{2.75}$$

  then $B(u,\varepsilon)$ is negligible in comparison to $A(u,\varepsilon)$ for all stationary points. Such terms are called globally secondary.

- Let $\breve{u}\to 0$ for $\varepsilon\to 0$, and for any sequence $\{u_n\}$ converging to $u=0$, if

$$\lim_{\substack{n\to\infty\\ \varepsilon\to 0}}\left|\frac{B(u,\eta)}{A(u,\eta)}\right|=0 \tag{2.76}$$

  then $B(u,\varepsilon)$ is negligible in comparison to $A(u,\varepsilon)$ for all stationary points $\breve{u}$. Likely, such terms are called locally secondary.

To be more specific, in the example discussed above, the term $u^2\varepsilon$ is globally secondary with respect to u^2, the term $u\varepsilon^2$ is globally secondary with respect to $2u\varepsilon$ while u^3 is locally secondary with respect to u^2 in the neighborhood of the point $u=0$.

Chapter 3

Homogenization for Composites with Finite Temperature Change

In this chapter, we will construct a new thermomechanical model for homogenizing heterogeneous materials made of temperature dependent constituents subject to finite temperature changes with the consideration that the total strain is still small based on the small-strain thermoelasticity theory developed by Kovalenko as introduced in Section 2.1.4. First we present the derivation of a Helmholtz free energy suitable for finite temperature changes, then we use the energy expression to construct a new thermomechanical micromechanics model, extending the previous work which was restricted to only small temperature change and temperature independent material properties. The new model is implemented using VAMUCH applying finite element method for the purpose of handling real heterogeneous materials with arbitrary microstructures.

Because of the restrictions of conventional linear thermoelasticity theory, this theory is barely applicable in the real cases of engineering studies. For instance, if a system starts at room temperature T_0, the current temperature is T after it reaches steady state, the ratio of temperature change $\frac{T-T_0}{T_0}$ must be below the level of elastic strain which is usually in the order of 1% or smaller to strictly satisfy the assumption of small strain changes adopted in conventional linear thermoelasticity. It is a very limiting assumption as many engineering systems are commonly designed to experience significant temperature changes of hundreds of degrees or even thousands of degrees such as space shuttle thermal protection panels, gas turbine blades, and car or airplane heat exchangers. Although the temperature change is large, the strains required to generate from these systems are still small in order to maintain the functionalities of systems. Hence, it has a practical significance for us to abandon the assumption of small temperature changes without violating the assumption of small strains

since the coefficients of thermal expansion (CTEs) for most materials are in the order of $10^{-6}K$. What worthy to mention is that we only need to slightly modify the well established linear thermoelasticity theory to enable such a generalization.

Also in linear thermoelasticity, it is also implicitly assumed that the material properties are temperature independent and the properties at the starting temperature T_0 are used directly in the calculation, while we also relax this assumption so that the material properties of the current temperature will be used in the calculation instead. We follow the small-strain thermoelasticity theory derived by Kovalenko and construct a Helmholtz free energy functional similar to that presented in Boussa [92].

3.1 Helmholtz Free Energy for Finite Temperature Change with Temperature Dependent Properties

The Helmholtz free energy functional is expanded into a quadratic form of strain field due to the assumption that the strain can be considered small and some remaining terms which are determined through the basic concepts of thermodynamics. To construct the formulation, we first define the material properties of interest as temperature dependent, such as the coefficients of thermal expansion $\alpha_{ij}(\sigma_{kl}, T)$, the heat capacity per unit volume $C_\varepsilon(\varepsilon_{ij}, T)$, the elastic constants $C_{ijkl}(T)$, the thermal strain tensors $m_{ij}(T)$, and the thermal stress tensors $l_{ij}(T)$. The symbol outside the parenthesis denotes the physical quantity while the symbols inside parenthesis are regarded as the independent variables used to describe the state of function. Note that for a defined function $F(\sigma_{ij}, T)$ or $F(\varepsilon_{ij}, T)$, the quantity $F(0, T)$ means $F(\sigma_{ij} = 0, T)$(stress free state) or $F(\varepsilon_{ij} = 0, T)$(strain free state) depending on how the function is defined.

The Helmholtz free energy density $f(\varepsilon_{ij}, T)$ is a function of the strain field ε_{ij} and the absolute temperature T. To relax the assumption of small temperature changes, we do not put any restriction on T but assuming ϵ_{ij} to be small, then we can carry out a Taylor

expansion of $f(\varepsilon_{ij}, T)$ in terms of the small strain field, ε_{ij}, as:

$$f(\varepsilon_{ij}, T) = f(0, T) + \varepsilon_{ij} \frac{\partial f(\varepsilon_{ij}, T)}{\partial \varepsilon_{ij}}|_{\varepsilon_{ij}=0} + \frac{1}{2}\varepsilon_{ij}\varepsilon_{kl} \frac{\partial^2 f(\varepsilon_{ij}, T)}{\partial \varepsilon_{ij} \partial \varepsilon_{kl}}|_{\varepsilon_{ij}=0} \tag{3.1}$$

Here only up to the quadratic terms of the strain field are kept due to the assumption of small strains. The terms of strain with order higher than quadratic will be extremely small so we neglect those terms. We know $\sigma_{ij} = \frac{\partial f}{\partial \varepsilon_{ij}}$, that is:

$$\sigma_{ij} = C_{ijkl}(T)\varepsilon_{kl} + l_{ij}(T) \tag{3.2}$$

with $C_{ijkl}(T) = \frac{\partial^2 f(\varepsilon_{ij}, T)}{\partial \varepsilon_{ij} \partial \varepsilon_{kl}}|_{\varepsilon_{ij}=0}$ as the fourth-order elasticity tensors and $l_{ij}(T) = \frac{\partial f(\varepsilon_{ij}, T)}{\partial \varepsilon_{ij}}|_{\varepsilon_{ij}=0}$ as the second-order thermal stress tensors. We can also rewrite the stress-strain relations as:

$$\varepsilon_{ij} = S_{ijkl}(T)\sigma_{kl} + m_{ij}(T) \tag{3.3}$$

with S_{ijkl} as the fourth-order compliance tensors and m_{ij} as the second-order thermal strain tensors which are obtained according to $m_{ij} = -S_{ijkl}l_{kl}$. The coefficients of thermal expansion, α_{ij}, as functions of stress field and temperature, are defined as:

$$\alpha_{ij} = \frac{\partial \varepsilon_{ij}}{\partial T}|_{\sigma_{ij}=\text{constant}} \tag{3.4}$$

Then from Eq. (3.3) and (3.4), we have:

$$\alpha_{ij} = S'_{ijkl}\sigma_{kl} + m'_{ij} \tag{3.5}$$

where prime is used to denote derivative with respect to T, the absolute temperature which is currently experienced by the solid, *i.e.*, $m'_{ij} = \frac{\mathrm{d}m_{ij}}{\mathrm{d}T}$. From Eq. (3.5), we have:

$$\alpha_{ij}(0, T) = m'_{ij} \tag{3.6}$$

where we can obtain:

$$m_{ij} = \int_{T_0}^{T} \alpha_{ij}(0,\zeta)\mathrm{d}\zeta + m_{ij}(T_0) \tag{3.7}$$

Note here $\alpha_{ij}(0,T)$ are the stress-free coefficients of thermal expansion which can be measured at a specific temperature T. The thermal strains at reference temperature, $m_{ij}(T_0)$, can be determined according to Eq. (3.3) if we know the initial stress and strain at the reference temperature. For example, if we choose our reference state to be stress and strain free at $T = T_0$, which is normally done, we will have $m_{ij}(T_0) = 0$. Then we can express our thermal strain tensor in a form similar as that we used in linear thermoelasticity which is restricted to only small temperature changes:

$$m_{ij} = \check{\alpha}_{ij}(T)\theta \quad \text{with} \quad \check{\alpha}_{ij}(T) = \frac{1}{\theta}\int_{T_0}^{T_0+\theta} \alpha_{ij}(0,\zeta)\mathrm{d}\zeta \tag{3.8}$$

where $\theta = T - T_0$ denotes the temperature change from the reference temperature T_0. It is emphatically pointed out that θ is not necessarily small comparing to T_0, as assumed in linear thermoelasticity. The thermal strain m_{ij} are not linear with respect to θ as $\check{\alpha}_{ij}(T)$ are also functions of θ (note $T = \theta + T_0$). We can observe from Eq. (3.8) that if the material properties are not functions of T, then the constitutive relation in Eq. (3.2) remains the same as that obtained in linear thermoelasticity. In other words, linear thermoelasticity is applicable to large temperature changes if the material properties are temperature independent. Normally, $\check{\alpha}_{ij}(T)$ are termed as the secant stress-free coefficients of thermal expansion. We can also express the thermal stress tensor as:

$$l_{ij}(T) = -C_{ijkl}(T)m_{kl}(T) = -C_{ijkl}(T)\check{\alpha}_{ij}(T)\theta \equiv \check{\beta}_{ij}(T)\theta \tag{3.9}$$

where $\check{\beta}_{ij}(T)$ can be similarly called the secant strain-free thermal stress coefficients.

Next we need to find the expression for $f(0,T)$. The entropy is commonly defined as $\eta = -\frac{\partial f}{\partial T}|_{\varepsilon_{ij}=\text{constant}}$ in continuum mechanics textbooks. From Eq. (3.1), we have:

$$\eta = -\frac{1}{2}C'_{ijkl}\varepsilon_{ij}\varepsilon_{kl} - l'_{ij}\varepsilon_{ij} - f'(0,T) \tag{3.10}$$

The heat capacity per unit volume at constant strain, C_ε, is defined as:

$$C_\varepsilon = T\frac{\partial \eta}{\partial T}|_{\varepsilon_{ij}=\text{constant}} \tag{3.11}$$

Based on this definition we have:

$$C_\varepsilon = -T\left[f''(0,T) + l''_{ij}\varepsilon_{ij} + \frac{1}{2}C''_{ijkl}\varepsilon_{ij}\varepsilon_{kl}\right] \tag{3.12}$$

Clearly C_ε is a function of both ε_{ij} and T, and the strain-free heat capacity per unit volume, $C_\varepsilon(0,T)$, will be a function of T only as follows:

$$C_\varepsilon(0,T) = -Tf''(0,T) \tag{3.13}$$

Dividing both sides of the above equation by T and integrating the result between T_0 and T twice yields:

$$f(0,T) = f_0 - \eta_0(T-T_0) - \int_{T_0}^{T}\int_{T_0}^{\zeta}\frac{C_\varepsilon(0,\rho)}{\rho}\mathrm{d}\rho\mathrm{d}\zeta \tag{3.14}$$

Substituting Eq. (3.9) and Eq. (3.14) into Eq. (3.1), we have:

$$f(\varepsilon_{ij},T) = \frac{1}{2}C_{ijkl}(T)\varepsilon_{ij}\varepsilon_{kl} + \check{\beta}_{ij}(T)\varepsilon_{ij}\theta - \eta_0 T - \int_{T_0}^{T}\int_{T_0}^{\zeta}\frac{C_\varepsilon(0,\rho)}{\rho}\mathrm{d}\rho\mathrm{d}\zeta \tag{3.15}$$

where the constant $f_0 + \eta_0 T_0$ is dropped as one can easily show this to be the internal energy of the reference state which is commonly assumed to be zero [91]. Although the free energy is linear with respect to η_0, entropy at reference temperature, it only provides an additive constant to the entropy at the current temperature and it has no effect on the thermoelastic behavior we want to model. Thus the term $\eta_0 T$ will be dropped in further derivations. The free energy form in Eq. (3.15) can be reduced to that used in [82] for micromechanics modeling based on linear thermoelasticity if we assume small temperature changes and the temperature independent material properties. This systematic derivation

using basic thermodynamics concepts above actually helped us identify an error in [82] that the sign of the quadratic terms related with temperature changes should be minus.

3.2 Micromechanics Model for Finite Temperature Change with Temperature Dependent Properties

Starting from the Helmholtz free energy expression we have just derived for finite temperature change small strain thermoelasticity in Eq. (3.15), we can follow an identical derivation procedure as given in [82] to obtain a variational statement which will govern the micromechanics model. To avoid repetition, this procedure is not reproduced here but suffice to say that the variational statement can be expressed as minimizing the following functional:

$$\begin{aligned} f(\bar{\varepsilon}_{ij},\chi_i,T) = \frac{1}{2\Omega}\int_\Omega \Big\{ & C_{ijkl}(T)\left[\bar{\varepsilon}_{ij}+\chi_{(i|j)}\right]\left[\bar{\varepsilon}_{kl}+\chi_{(k|l)}\right] \\ & + 2\check{\beta}_{ij}(T)\left[\bar{\varepsilon}_{ij}+\chi_{(i|j)}\right]\theta\Big\}\mathrm{d}\Omega - \int_{T_0}^{T}\int_{T_0}^{\zeta}\frac{\langle C_\varepsilon(0,\rho)\rangle}{\rho}\mathrm{d}\rho\mathrm{d}\zeta \end{aligned} \tag{3.16}$$

subject to periodic constraints. Here, χ_i are the commonly called fluctuating functions, $\bar{\varepsilon}_{ij}$ are the global strain tensors, and angle brackets indicate average over the unit cell.

Introduce the following matrix notations:

$$\bar{\varepsilon} = \lfloor \bar{\varepsilon}_{11} \;\; 2\bar{\varepsilon}_{12} \;\; \bar{\varepsilon}_{22} \;\; 2\bar{\varepsilon}_{13} \;\; 2\bar{\varepsilon}_{23} \;\; \bar{\varepsilon}_{33} \rfloor^T \tag{3.17}$$

$$\begin{Bmatrix} \frac{\partial\chi_1}{\partial y_1} \\ \frac{\partial\chi_1}{\partial y_2}+\frac{\partial\chi_2}{\partial y_1} \\ \frac{\partial\chi_2}{\partial y_2} \\ \frac{\partial\chi_1}{\partial y_3}+\frac{\partial\chi_3}{\partial y_1} \\ \frac{\partial\chi_2}{\partial y_3}+\frac{\partial\chi_3}{\partial y_2} \\ \frac{\partial\chi_3}{\partial y_3} \end{Bmatrix} = \begin{bmatrix} \frac{\partial}{\partial y_1} & 0 & 0 \\ \frac{\partial}{\partial y_2} & \frac{\partial}{\partial y_1} & 0 \\ 0 & \frac{\partial}{\partial y_2} & 0 \\ \frac{\partial}{\partial y_3} & 0 & \frac{\partial}{\partial y_1} \\ 0 & \frac{\partial}{\partial y_3} & \frac{\partial}{\partial y_2} \\ 0 & 0 & \frac{\partial}{\partial y_3} \end{bmatrix} \begin{Bmatrix} \chi_1 \\ \chi_2 \\ \chi_3 \end{Bmatrix} \equiv \Gamma_h\chi \tag{3.18}$$

where Γ_h is an operator matrix and χ is a column matrix containing the three components of the fluctuating functions. If we discretize χ using the finite elements as:

$$\chi(x_i; y_i) = S(y_i)\mathcal{X}(x_i) \tag{3.19}$$

where S represents the shape functions and $\mathcal{X}$ is a column matrix of the nodal values of the fluctuation functions. Substituting Eqs. (3.17), (3.18), and (3.19) into Eq. (3.16), we obtain a discretized version of the functional as:

$$\begin{aligned} f(\bar{\varepsilon}, \mathcal{X}, T) = \frac{1}{2\Omega}(&\mathcal{X}^T E \mathcal{X} + 2\mathcal{X}^T D_{h\varepsilon}\bar{\varepsilon} + \bar{\varepsilon}^T D_{\varepsilon\varepsilon}\bar{\varepsilon} \\ &+ 2\mathcal{X}^T D_{h\theta}\theta + 2\bar{\varepsilon}^T D_{\varepsilon\theta}\theta) - \int_{T_0}^{T}\int_{T_0}^{\zeta} \frac{\langle C_\varepsilon(0,\rho)\rangle}{\rho}\mathrm{d}\rho\mathrm{d}\zeta \end{aligned} \tag{3.20}$$

where

$$\begin{aligned} E &= \int_\Omega (\Gamma_h S)^T D(\Gamma_h S)\mathrm{d}\Omega & D_{h\varepsilon} &= \int_\Omega (\Gamma_h S)^T D\mathrm{d}\Omega \\ D_{\varepsilon\varepsilon} &= \int_\Omega D\mathrm{d}\Omega & D_{h\theta} &= \int_\Omega (\Gamma_h S)^T \check{\beta}\mathrm{d}\Omega \\ D_{\varepsilon\theta} &= \int_\Omega \check{\beta}\mathrm{d}\Omega \end{aligned}$$

with D as the 6×6 material matrix condensed from the fourth order elasticity tensor C_{ijkl}, and $\check{\beta}$ as the 6×1 column condensed from $\check{\beta}_{ij}$. Minimizing $f(\bar{\varepsilon}, \mathcal{X}, T)$ in Eq. (3.20) with respect to $\mathcal{X}$, we obtain the following linear system:

$$E\mathcal{X} = -D_{h\varepsilon}\bar{\varepsilon} - D_{h\theta}\theta \tag{3.21}$$

The solution can be written symbolically as:

$$\mathcal{X} = \mathcal{X}_0\bar{\varepsilon} + \mathcal{X}_\theta\theta \tag{3.22}$$

Substituting Eq. (3.22) into Eq. (3.20), we can calculate the effective Helmholtz free energy density of the UC as:

$$\bar{f}(\bar{\varepsilon}, T) = \frac{1}{2}\bar{\varepsilon}^T \bar{D}(T)\bar{\varepsilon} + \bar{\varepsilon}^T \beta^*(T)\theta + \bar{f}(0, T) \tag{3.23}$$

with

$$\begin{aligned}
\bar{D}(T) &= \frac{1}{\Omega}(\mathcal{X}_0^T D_{h\varepsilon} + D_{\varepsilon\varepsilon}) \\
\beta^*(T) &= \frac{1}{\Omega}\left[\frac{1}{2}(D_{h\varepsilon}^T \mathcal{X}_\theta + \mathcal{X}_0^T D_{h\theta}) + D_{\varepsilon\theta}\right] \\
\bar{f}(0, T) &= \frac{\theta^2}{2\Omega}\mathcal{X}_\theta^T D_{h\theta} - \int_{T_0}^{T}\int_{T_0}^{\zeta} \frac{\langle C_\varepsilon(0, \rho)\rangle}{\rho} \mathrm{d}\rho \mathrm{d}\zeta
\end{aligned}$$

Here, we can observe that $\bar{D}$ is the effective stiffness matrix and $\bar{f}(0, T)$ is the effective heat capacity per unit volume when the temperature of the unit cell is increased from T_0 to T. However, β^* cannot be simply interpreted as the effective thermal stress coefficient matrix and its real meaning will be disclosed later. Comparing to the micromechanics model based on linear thermoelasticity, we will find out that the calculation and results of $\bar{D}$ remain the same as long as we use temperature dependent material properties for the computation and the calculation of β^* will remain the same if we replace the temperature independent CTE used for linear thermoelasticity with secant CTE, but the results will be obviously different.

The effective stress-strain relationship for the homogenized material can be written as:

$$\bar{\sigma} = \bar{D}\bar{\varepsilon} + \beta^*\theta \tag{3.24}$$

The effective thermal stress coefficient can be defined as follows:

$$\bar{\beta} = \frac{\partial \bar{\sigma}}{\partial T}|_{\sigma=\text{constant}} = \bar{D}'\bar{\varepsilon} + \beta^{*'}\theta + \beta^* \tag{3.25}$$

The effective thermal stress coefficient is a function of the global strain $\bar{\varepsilon}$ and absolute

temperature T. Note $\bar{\beta}$ is not the same as β^*. The corresponding effective strain-free thermal stress coefficient is:

$$\bar{\beta}(0,T) = \beta^{*'}(T - T_0) + \beta^* \tag{3.26}$$

The effective strain-stress relationship for the homogenized material can be obtained from Eq. (3.24) as:

$$\bar{\varepsilon} = \bar{D}^{-1}\bar{\sigma} - \bar{D}^{-1}\beta^*\theta \tag{3.27}$$

which implies the effective thermal strain, $\bar{m}$, can be obtained using the following expression:

$$\bar{m} = -\bar{D}^{-1}\beta^*\theta = \alpha^*\theta \tag{3.28}$$

If one would like to obtain the effective CTEs, we can obtain through its definition in Eq. (3.4) as:

$$\bar{\alpha}(\sigma,T) = \left(\bar{D}^{-1}\right)'\bar{\sigma} + \bar{m}' \tag{3.29}$$

where $\bar{m}'$ can be considered as the effective stress-free CTE at T, $\bar{\alpha}(0,T)$. Particularly, using Eq. (3.28), we have:

$$\bar{\alpha}(0,T) = -\bar{D}^{-1}\left(\beta^* + \beta^{*'}\theta - \bar{D}'\bar{D}^{-1}\beta^*\theta\right) = \alpha^* + \alpha^{*'}\theta \tag{3.30}$$

where the identity $\left(\bar{D}^{-1}\right)' = -\bar{D}^{-1}\bar{D}'\bar{D}^{-1}$ is used.

The effective specific heat per unit volume can also be obtained through its definition as:

$$\bar{C}_\varepsilon(\bar{\varepsilon},T) = -T\frac{\partial^2\bar{f}}{\partial T^2}|_{\bar{\varepsilon}=\text{constant}} = \bar{C}_\varepsilon(0,T) - T\left(\frac{1}{2}\bar{\varepsilon}^T\bar{D}''\bar{\varepsilon} + \bar{\varepsilon}^T(\beta^*\theta)''\right) \tag{3.31}$$

with

$$\bar{C}_\varepsilon(0,T) = \langle C_\varepsilon(0,T)\rangle - T\left(\frac{\theta^2}{2\Omega}\mathcal{X}_\theta^T D_{h\theta}\right)'' \tag{3.32}$$

as the strain-free, effective specific heat per unit volume. Let $F(T) = \frac{\mathcal{X}_\theta^T D_{h\theta}}{\Omega}$, $\bar{C}_\varepsilon(0,T)$ can be evaluated as:

$$\bar{C}_\varepsilon(0,T) = \langle C_\varepsilon(0,T)\rangle - TF - 2T\theta F' - \frac{T\theta^2}{2}F'' \tag{3.33}$$

Usually, we desire to use $\bar{D}$, $\bar{\alpha}(0,T)$, and $\bar{C}_\varepsilon(0,T)$ to characterize the macroscopic thermoelastic behavior of the heterogeneous materials. The computation of $\bar{\alpha}(0,T)$ and $\bar{C}_\varepsilon(0,T)$ requires the derivatives of $\bar{D}$, β^*, and F with respect to temperature, which implies we need to differentiate Eq. (3.21) with respect to temperature such as:

$$E'\mathcal{X} + E\mathcal{X}' = -D'_{h\varepsilon}\bar{\varepsilon} - D'_{h\theta} \tag{3.34}$$

This equation can be used to solve for $\mathcal{X}' = \mathcal{X}'_0\bar{\varepsilon} + \mathcal{X}'_\theta$ once $\mathcal{X}$ has been solved from the original equation in Eq. (3.21). The second derivatives can be evaluated similarly. Although feasible, this approach introduces unwarranted complexity and longer computing time in real applications. A much more practical and simpler approach is to fit the values of α^*, F with respect to T as a simple function such as a polynomial, then evaluate the needed first derivative of α^* to obtain $\bar{\alpha}(0,T)$ and evaluate the needed first and second derivatives of F to obtain $\bar{C}_\varepsilon(0,T)$. This approach also allows us to reuse the VAMUCH code developed in [82] to implement the present theory with minor changes.

It is worthy to point out that if one assumes that the constituent material properties are temperature independent, that is:

$$C'_{ijkl} = 0 \qquad \check{\alpha}'_{ij} = 0 \tag{3.35}$$

then we have:

$$\bar{\beta}(\bar{\varepsilon},T) = \bar{\beta}(0,T) = \beta^* \tag{3.36}$$

$$\bar{\alpha}(\bar{\varepsilon},T) = \bar{\alpha}(0,T) = -\bar{D}^{-1}\beta^* \tag{3.37}$$

$$\bar{C}_\varepsilon(\bar{\varepsilon},T) = \bar{C}_\varepsilon(0,T) = \langle C_\varepsilon(0,T)\rangle - TF \tag{3.38}$$

These formulas are exactly the same as those in [82] if we realize that work further restricts small temperature variations, which implies that T can be replaced by T_0. Note the sign difference before TF because the sign in front of the energy term related with specific heat should be minus in Eq. (2) of [82].

After having obtained the effective material properties, we can use them to carry out various macroscopic thermoelastic analyses of the homogenized effective medium under different loading and temperature conditions, output of which should be global displacements and strains.

If the local fields within the UC are of interest, we can recover those fields after we have obtained the macroscopic behavior which can be described by global displacements v_i and global strains $\bar{\varepsilon}$ [82].

$$u = v + \begin{bmatrix} \frac{\partial v_1}{\partial x_1} & \frac{\partial v_1}{\partial x_2} & \frac{\partial v_1}{\partial x_3} \\ \frac{\partial v_2}{\partial x_1} & \frac{\partial v_2}{\partial x_2} & \frac{\partial v_2}{\partial x_3} \\ \frac{\partial v_3}{\partial x_1} & \frac{\partial v_3}{\partial x_2} & \frac{\partial v_3}{\partial x_3} \end{bmatrix} \begin{Bmatrix} y_1 \\ y_2 \\ y_3 \end{Bmatrix} + \begin{Bmatrix} \chi_1 \\ \chi_2 \\ \chi_3 \end{Bmatrix} \tag{3.39}$$

with u as the column matrix of u_i and v as the column matrix of v_i. The local strain field can be recovered using:

$$\varepsilon = \bar{\varepsilon} + \Gamma_h \chi \tag{3.40}$$

Finally, the local stress field can be recovered straightforwardly using the 3D constitutive relations for the constituent material as:

$$\sigma = D\varepsilon + \check{\beta}\theta \tag{3.41}$$

3.3 Validation using an Analytical Solution for Binary Composites

To demonstrate and validate the predictability and capability of VAMUCH, we consider a periodic binary composite formed by orthotropic layers and the material axes are the same as the global coordinates x_i so that the material is uniform in the x_1 - x_2 plane and periodic along x_3 direction. A typical unit cell can be identified as shown in Fig. 3.1, the dimension

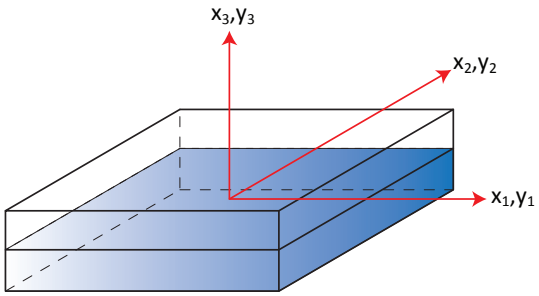

Fig. 3.1: Sketch of a binary composite

along y_3 is h and dimensions along y_1 and y_2 can be arbitrary. Let ϕ_1 and ϕ_2 denote the volume fractions of the first layer and the second layer, respectively, and we have $\phi_1+\phi_2 = 1$.

Because the material is uniform in the x_1 - x_2 plane, the fluctuating function χ_i will be a function of only y_3. By using the technique of Lagrange multipliers, the variational statement of micromechanical analysis of UC can be posed as:

$$\begin{aligned} J =& \frac{1}{2h}\left[\int_{-\frac{h}{2}}^{(\phi_1-\frac{1}{2})h}\left(\varepsilon^{(1)T}D^{(1)}\varepsilon^{(1)}+2\varepsilon^{(1)T}\breve{\beta}^{(1)}\theta\right)\mathrm{d}y_3\right. \\ &\left.+\int_{(\frac{1}{2}-\phi_2)h}^{\frac{h}{2}}\left(\varepsilon^{(2)T}D^{(2)}\varepsilon^{(2)}+2\varepsilon^{(2)T}\breve{\beta}^{(2)}\theta\right)\mathrm{d}y_3\right]+\lambda_i\left\langle\chi_i\right\rangle \\ &+\beta_{i3}\left[\chi_i^{(2)}(y_3,\frac{h}{2})-\chi_i^{(1)}(y_3,-\frac{h}{2})\right]-\int_{T_0}^{T}\int_{T_0}^{\zeta}\frac{\langle C_\varepsilon(0,\rho)\rangle}{\rho}\mathrm{d}\rho\mathrm{d}\zeta \end{aligned} \tag{3.42}$$

with

$$\varepsilon^{(\alpha)} = \lfloor \bar{\varepsilon}_{11}\ \ 2\bar{\varepsilon}_{12}\ \ \bar{\varepsilon}_{22}\ \ 2\bar{\varepsilon}_{13}+\frac{\partial\chi_1^{(\alpha)}}{\partial y_3}\ \ 2\bar{\varepsilon}_{23}+\frac{\partial\chi_2^{(\alpha)}}{\partial y_3}\ \ \bar{\varepsilon}_{33}+\frac{\partial\chi_3^{(\alpha)}}{\partial y_3}\rfloor^T$$

$$\breve{\beta}^{(\alpha)} = \lfloor \breve{\beta}_{11}^{(\alpha)}\ \ \breve{\beta}_{12}^{(\alpha)}\ \ \breve{\beta}_{22}^{(\alpha)}\ \ \breve{\beta}_{13}^{(\alpha)}\ \ \breve{\beta}_{23}^{(\alpha)}\ \ \breve{\beta}_{33}^{(\alpha)}\rfloor^T$$

where $\alpha = 1, 2$ denote two layers and $\chi^{(\alpha)}$ are the fluctuating functions for two layers. The material matrices $D^{(\alpha)}$ are characterized by the nine constants for the orthotropic elastic

materials and arranged as:

$$D^{(\alpha)} = \begin{bmatrix} c_{11}^{(\alpha)} & 0 & c_{12}^{(\alpha)} & 0 & 0 & c_{13}^{(\alpha)} \\ 0 & c_{66}^{(\alpha)} & 0 & 0 & 0 & 0 \\ c_{12}^{(\alpha)} & 0 & c_{22}^{(\alpha)} & 0 & 0 & c_{23}^{(\alpha)} \\ 0 & 0 & 0 & c_{55}^{(\alpha)} & 0 & 0 \\ 0 & 0 & 0 & 0 & c_{44}^{(\alpha)} & 0 \\ c_{13}^{(\alpha)} & 0 & c_{23}^{(\alpha)} & 0 & 0 & c_{33}^{(\alpha)} \end{bmatrix} \tag{3.43}$$

Following the normal procedure of calculus of variations, we can solve for the fluctuation functions which can be used to obtain the following effective Helmholtz free energy as:

$$\Pi_\Omega = \frac{1}{2} \begin{Bmatrix} \bar{\varepsilon}_{11} \\ 2\bar{\varepsilon}_{12} \\ \bar{\varepsilon}_{22} \\ 2\bar{\varepsilon}_{13} \\ 2\bar{\varepsilon}_{23} \\ \bar{\varepsilon}_{33} \end{Bmatrix}^T \begin{bmatrix} c_{11}^* & 0 & c_{12}^* & 0 & 0 & c_{13}^* \\ 0 & c*_{66} & 0 & 0 & 0 & 0 \\ c_{12}^* & 0 & c_{22}^* & 0 & 0 & c_{23}^* \\ 0 & 0 & 0 & c_{55}^* & 0 & 0 \\ 0 & 0 & 0 & 0 & c_{44}^* & 0 \\ c_{13}^* & 0 & c_{23}^* & 0 & 0 & c_{33}^* \end{bmatrix} \begin{Bmatrix} \bar{\varepsilon}_{11} \\ 2\bar{\varepsilon}_{12} \\ \bar{\varepsilon}_{22} \\ 2\bar{\varepsilon}_{13} \\ 2\bar{\varepsilon}_{23} \\ \bar{\varepsilon}_{33} \end{Bmatrix} + \begin{Bmatrix} \bar{\varepsilon}_{11} \\ 2\bar{\varepsilon}_{12} \\ \bar{\varepsilon}_{22} \\ 2\bar{\varepsilon}_{13} \\ 2\bar{\varepsilon}_{23} \\ \bar{\varepsilon}_{33} \end{Bmatrix}^T \begin{Bmatrix} \beta_{11}^* \\ \beta_{12}^* \\ \beta_{22}^* \\ \beta_{13}^* \\ \beta_{23}^* \\ \beta_{33}^* \end{Bmatrix} \theta + f^* \tag{3.44}$$

It can be observed that the homogenized material properties still possess the same orthotropic symmetry for this binary composite case, although in general the homogenized material could be general anisotropic, which means a fully populated 6×6 stiffness matrix.

The expressions of effective elastic properties c^*_{ij} are listed below:

$$
\begin{aligned}
c^*_{11} &= \langle c_{11} \rangle - \frac{\phi_1 \phi_2 (c^{(2)}_{13} - c^{(1)}_{13})^2}{\phi_1 c^{(2)}_{33} + \phi_2 c^{(1)}_{33}} \\
c^*_{12} &= \langle c_{12} \rangle - \frac{\phi_1 \phi_2 (c^{(2)}_{13} - c^{(1)}_{13})(c^{(2)}_{23} - c^{(1)}_{23})}{\phi_1 c^{(2)}_{33} + \phi_2 c^{(1)}_{33}} \\
c^*_{13} &= \frac{\phi_1 c^{(1)}_{13} c^{(2)}_{33} + \phi_2 c^{(2)}_{13} c^{(1)}_{33}}{\phi_1 c^{(2)}_{33} + \phi_2 c^{(1)}_{33}} \\
c^*_{22} &= \langle c_{22} \rangle - \frac{\phi_1 \phi_2 (c^{(2)}_{23} - c^{(1)}_{23})^2}{\phi_1 c^{(2)}_{33} + \phi_2 c^{(1)}_{33}} \\
c^*_{23} &= \frac{\phi_1 c^{(1)}_{23} c^{(2)}_{33} + \phi_2 c^{(2)}_{23} c^{(1)}_{33}}{\phi_1 c^{(2)}_{33} + \phi_2 c^{(1)}_{33}} \\
c^*_{33} &= 1 / \left\langle \frac{1}{c_{33}} \right\rangle \quad c^*_{44} = 1 / \left\langle \frac{1}{c_{44}} \right\rangle \\
c^*_{55} &= 1 / \left\langle \frac{1}{c_{55}} \right\rangle \quad c^*_{66} = \langle c_{66} \rangle
\end{aligned}
\tag{3.45}
$$

The thermal stress coefficients β^*_{ij} can be expressed as:

$$
\begin{aligned}
\beta^*_{11} &= \langle \check{\beta}_{11} \rangle - \frac{\phi_1 \phi_2 (c^{(1)}_{13} - c^{(2)}_{13})(\check{\beta}^{(1)}_{33} - \check{\beta}^{(2)}_{33})}{\phi_1 c^{(2)}_{33} + \phi_2 c^{(1)}_{33}} \\
\beta^*_{12} &= \langle \check{\beta}_{12} \rangle \\
\beta^*_{22} &= \langle \check{\beta}_{22} \rangle - \frac{\phi_1 \phi_2 (c^{(1)}_{23} - c^{(2)}_{23})(\check{\beta}^{(1)}_{33} - \check{\beta}^{(2)}_{33})}{\phi_1 c^{(2)}_{33} + \phi_2 c^{(1)}_{33}} \\
\beta^*_{13} &= \frac{(c^{(2)}_{55} \check{\beta}^{(1)}_{13} \phi_1 + c^{(1)}_{55} \check{\beta}^{(2)}_{13} \phi_2)}{\phi_1 c^{(2)}_{55} + \phi_2 c^{(1)}_{55}} \\
\beta^*_{23} &= \frac{(c^{(2)}_{44} \check{\beta}^{(1)}_{23} \phi_1 + c^{(1)}_{44} \check{\beta}^{(2)}_{23} \phi_2)}{\phi_1 c^{(2)}_{44} + \phi_2 c^{(1)}_{44}} \\
\beta^*_{33} &= \frac{(c^{(2)}_{33} \check{\beta}^{(1)}_{33} \phi_1 + c^{(1)}_{33} \check{\beta}^{(2)}_{33} \phi_2)}{\phi_1 c^{(2)}_{33} + \phi_2 c^{(1)}_{33}}
\end{aligned}
\tag{3.46}
$$

The effective heat $\bar{f}(0,T)$ of the binary composite can be calculated as:

$$\bar{f}(0,T) = -\int_{T_0}^{T}\int_{T_0}^{\zeta}\frac{\langle C_\epsilon(0,\rho)\rangle}{\rho}\mathrm{d}\rho\mathrm{d}\zeta - \phi_1\phi_2\theta^2\left[\frac{(\check{\beta}_{33}^{(1)}-\check{\beta}_{33}^{(2)})^2}{\phi_1 c_{33}^{(2)}+\phi_2 c_{33}^{(1)}}+\frac{(\check{\beta}_{23}^{(1)}-\check{\beta}_{23}^{(2)})^2}{\phi_1 c_{44}^{(2)}+\phi_2 c_{44}^{(1)}}+\frac{(\check{\beta}_{13}^{(1)}-\check{\beta}_{13}^{(2)})^2}{\phi_1 c_{55}^{(2)}+\phi_2 c_{55}^{(1)}}\right] \tag{3.47}$$

These analytical expressions of the binary composite example can be used to validate the general-purpose micromechanics code VAMUCH for its capability in modeling heterogeneous materials made of temperature dependent constituents and subjected to finite temperature changes.

3.4 Numerical Examples

Several numerical examples including binary composites, fiber reinforced composites, and particle reinforced composites are used to validate and demonstrate the new capability based on the present model implemented in VAMUCH. The differences between VAMUCH based on linear thermoelasticity and VAMUCH based on finite temperature change small strain thermoelasticity for predicting effective properties including effective CTEs and specific heats, and local fields will be carefully quantified. Without loss of generality, we assume that the two constituents in composites are isotropic with temperature dependent material properties including Young's modulus $E(T)$, Poisson's ratio $\nu(T)$, stress-free CTEs $\alpha(0,T)$, and strain-free specific heat $C_\epsilon(0,T)$ given in Table 3.1 and Table 3.2. Here we obtain the effective stress-free CTEs and effective strain-free specific heat by fitting the values of α^*, F with respect to T and evaluating the needed first derivatives and second derivatives according to Eq. (3.30) and Eq. (3.33).

3.4.1 Binary Composites

Let us first consider a binary composite with the bottom layer made of constituent 1 and the top layer made of constituent 2. The volume fraction of bottom layer is 0.3. Duocel silicon carbide form (8% norminal density) is used as our constituent 1 with stress-free CTEs

Table 3.1: Material property of constituent 1

T (°C)	E (GPa)	ν	α (μ/°C)	C_ϵ (KJ/m$^3-$°C)
23	2.76	0.22	1.22	249.4
50	2.76	0.22	1.37	250.9
75	2.76	0.22	1.51	251.9
100	2.76	0.22	1.65	253.2
125	2.76	0.22	1.78	254.2
150	2.76	0.22	1.90	255.3
175	2.76	0.22	1.98	255.8
200	2.76	0.22	2.06	256.3
225	2.76	0.22	2.13	256.8
250	2.76	0.22	2.18	257.3
275	2.76	0.22	2.23	257.6
300	2.76	0.22	2.28	257.8

Table 3.2: Material property of constituent 2

T (°C)	E (GPa)	ν	α (μ/°C)	C_ϵ (KJ/m$^3-$°C)
23	4.10	0.3	7.46	2280
50	3.57	0.3	8.13	2280
75	3.38	0.3	8.42	2280
100	3.25	0.3	8.45	2280
125	3.14	0.3	8.38	2280
150	3.05	0.3	8.28	2280
175	2.98	0.3	8.09	2280
200	2.92	0.3	7.89	2280
225	2.87	0.3	7.64	2280
250	2.81	0.3	7.38	2280
275	2.77	0.3	7.05	2280
300	2.72	0.3	6.70	2280

and strain-free specific heat obtained by curve fitting and Young's modulus directly taken based on existing available data from online resource [95]. Data of constituent 2 (thermoset phenolic resin matrix composite with glass-cloth-fabric reinforcements) are based on Table 7 and curve fitting of Fig. 47 and 50 of ASM handbook [96]. Using VAMUCH, this composite can be modeled using either a 1D UC, or 2D UC, or 3D UC. The reason is that the dimensionality of the problem necessary for VAMUCH analysis is completely determined by its periodicity. Binary composite has a 1D periodicity. Hence 1D UC is sufficient and although using higher dimensional models (2D UC or 3D UC) can also reproduce the same results, it is a unnecessary waste of computing time. Nevertheless, it serves as a good validation test of the VAMUCH to demonstrate it will compute according to its underlining theory. We verified that indeed 1D UC, 2D UC, and 3D UC predict exactly the same results, which is also exactly the same as the exact solution derived in the previous section.

In Fig. 3.2, we plot the Young's modulus variation with respect to temperature including in-plane modulus, transverse modulus, and the constituent moduli. We can observe as a composite, its Young's modulus having a temperature dependent behavior different from that of the constituents. Note the temperature dependent elastic constants will remain the same no matter whether the theory assumes small temperature changes or not. However, it is not true for CTEs. The temperature dependent CTEs of the binary composite are shown in Fig. 3.3 and Fig. 3.4, where small temperature change denotes the results based on linear thermoelasticity assuming small temperature changes while finite temperature change is based on the finite temperature change small strain thermoelasticity theory presented in this paper. As one can observe from both figures, the effective CTEs based on small temperature change assumptions vary more significantly with the temperature. One might argue that in the thermoelastic analysis, one should not directly use the effective CTEs calculated based on small temperature change assumptions, but use the secant CTEs defined from these temperature dependent effective CTEs. In other words, we use linear thermoelasticity for micromechanics modeling but finite temperature change small strain thermoelasticity for macroscopic stress analysis. For this reason, we also plot the effective

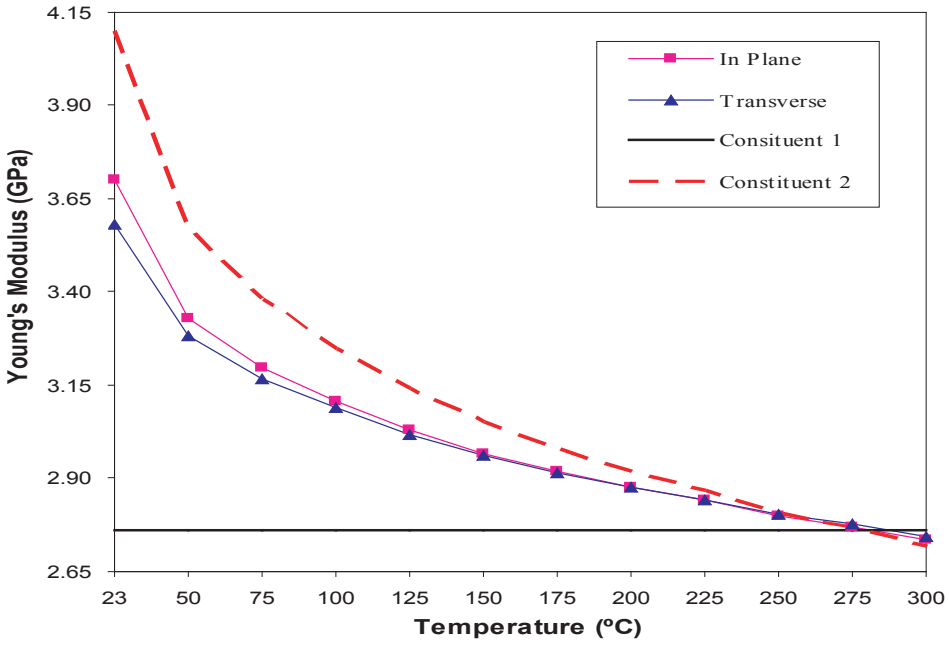

Fig. 3.2: Young's modulus variation with respect to temperature

secant CTEs based on assuming small temperature changes only for the micromechanics modeling. We observe that results predicted from finite temperature change small strain thermoelasticity theory are noticeably different (more than 10%) from the effective CTEs predicted with assuming small temperature changes and even more significantly different with results based on assuming small temperature changes only for the micromechanics modeling, which implies that the micromechanics model based on finite temperature change thermoelasticity theory is necessary to avoid loss of accuracy for large temperature changes. As far as the specific heat $\bar{C}_\epsilon$ concerned, as shown in Fig. 3.5, there are not much differences (less than 0.5%) between assuming small temperature changes or not. The main reason is that the major contribution comes from the specific heat of the constituents $\langle C_\epsilon(0,T)\rangle$ which is not affected by the limiting assumptions of linear thermoelasticity. Hypothetically speaking, according to Eq. (3.33), if F and its derivatives F' and F'' are not that small (F is in the order of 0.1 J/m$^3-^\circ$C^2 for this case), then the contribution from the last three terms can easily overpower the first term when T and thus θ is large.

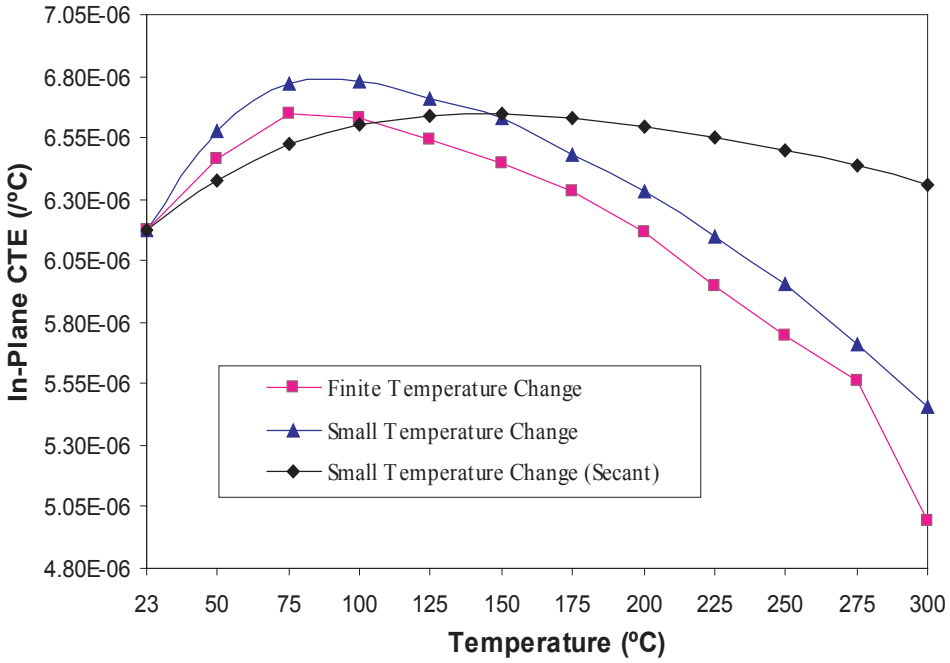

Fig. 3.3: In-plance CTE ($\bar{\alpha}_{11}, \bar{\alpha}_{22}$) change with respect to temperature

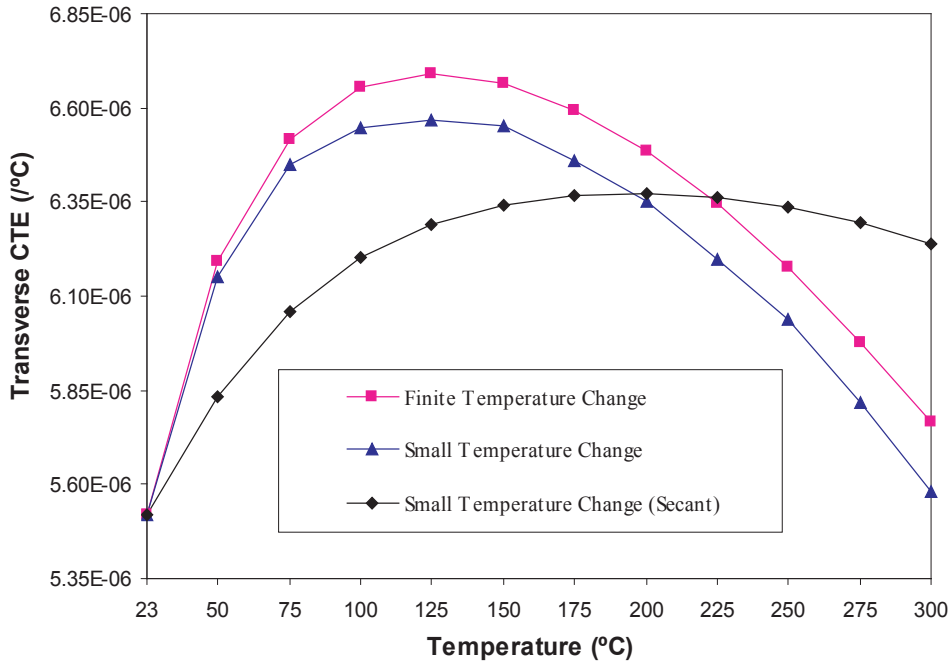

Fig. 3.4: Transverse CTE ($\bar{\alpha}_{33}$) change with respect to temperature

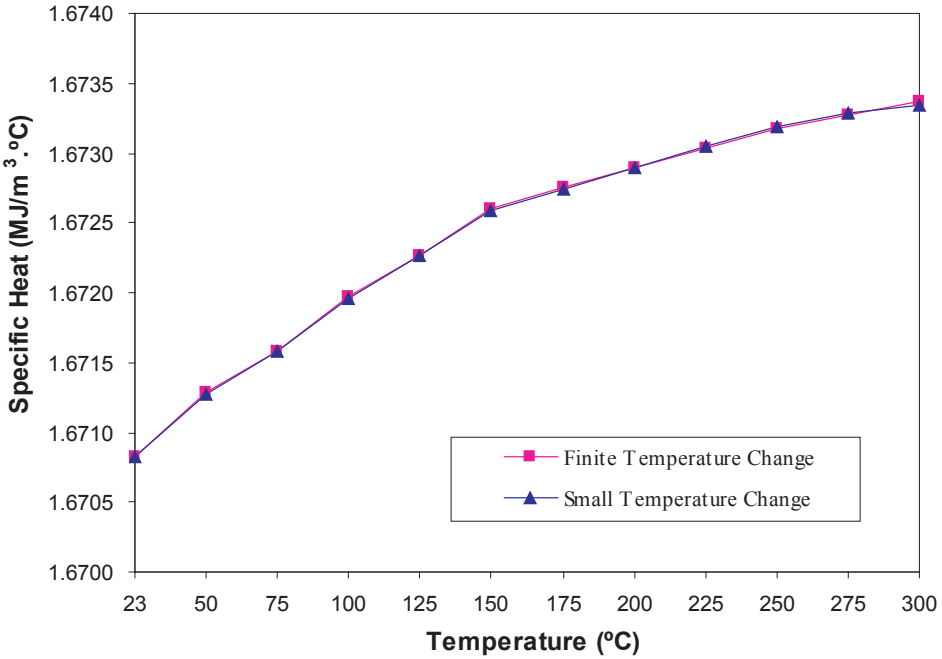

Fig. 3.5: Effective specific heat change with respect to temperature

3.4.2 Fiber Reinforced Composites

Now, let us consider a fiber reinforced composite with the constituent 1 as the fiber and constituent 2 as the matrix. The fiber volume fraction is kept 0.3 in order to compare with the other cases without the unnecessary involvement of the volume fraction factor. The microstructure is periodic in two dimensions and thus can be modeled using either 2D UC or 3D UC. 2D UC is the obvious choice as it will predict the same results as 3D UC with much less computation. For this fiber reinforced composite, we compare the effective coefficients of thermal expansion in longitudinal and transverse directions. From the results plotted in Fig. 3.6 and Fig. 3.7, we again verified that there are significant differences between the predictions based on finite temperature changes and those based on assuming small temperature changes. However, for this fiber reinforced composites, we notice that the CTEs computed from the effective CTEs obtained assuming small temperature changes are closer to those predicted using finite temperature changes small strain thermoelasticity. Particularly, they are very close to each other in the transverse direction. The specific heat

results still remain close to each other as what we have observed for the binary composite.

3.4.3 Particle Reinforced Composites

To validate and demonstrate the capability of the present theory as implemented in VAMUCH in modeling 3D microstructures, we choose a particle reinforced composite which should be modeled using a 3D unit cell. The particle is made of constituent 1 and has a volume fraction of 0.3. As expected, VAMUCH predicts the particle reinforced composite to be macroscopically isotropic, which is a commonly accepted fact. We plot the effective coefficient of thermal expansion $\bar{\alpha}$ predicted by VAMUCH in Fig. 3.8. Again, the results predicted using finite temperature changes are quite different from secant CTE computed from effective CTE predicted based on the assumption of small temperature changes. However, effective CTE obtained by assuming small temperature change is close to the effective CTE predicted using the finite temperature change small strain thermoelasticity theory, which was also observed for the transverse CTE for the fiber reinforced composites in Fig. 3.7.

3.4.4 Predict Local Stresses

The ultimate purpose of micromechanics is to reduce the original prohibitive computation of directly carrying out the macroscopic analysis of the structure with all the microstructural details without significant loss of accuracy. To achieve this, we first need to replace the original heterogeneous material with an imaginary homogeneous material with the effective properties predicted using a micromechanics model. Then we can carry out a much simpler structural analysis with homogenized material properties to obtain the global behavior. Most of micromechanics modeling efforts stop here. In fact, we also need to accurately compute the local fields within the microstructure based on the global behavior, particularly, if we want to study the failure of heterogeneous materials. To complete the modeling process, micromechanics models should also be able to predict the local fields based on a certain macroscopic field, which is called micromechanical recovery procedure in VAMUCH. To demonstrate the capability of our model in predicting local fields, we use

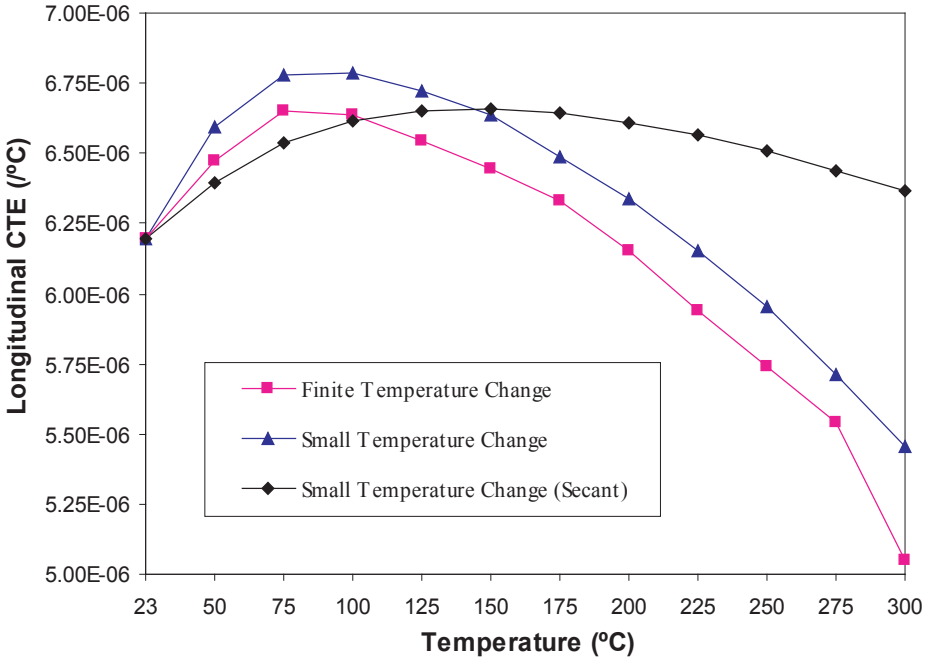

Fig. 3.6: Longitudinal CTE ($\bar{\alpha}_{11}$) change with respect to temperature

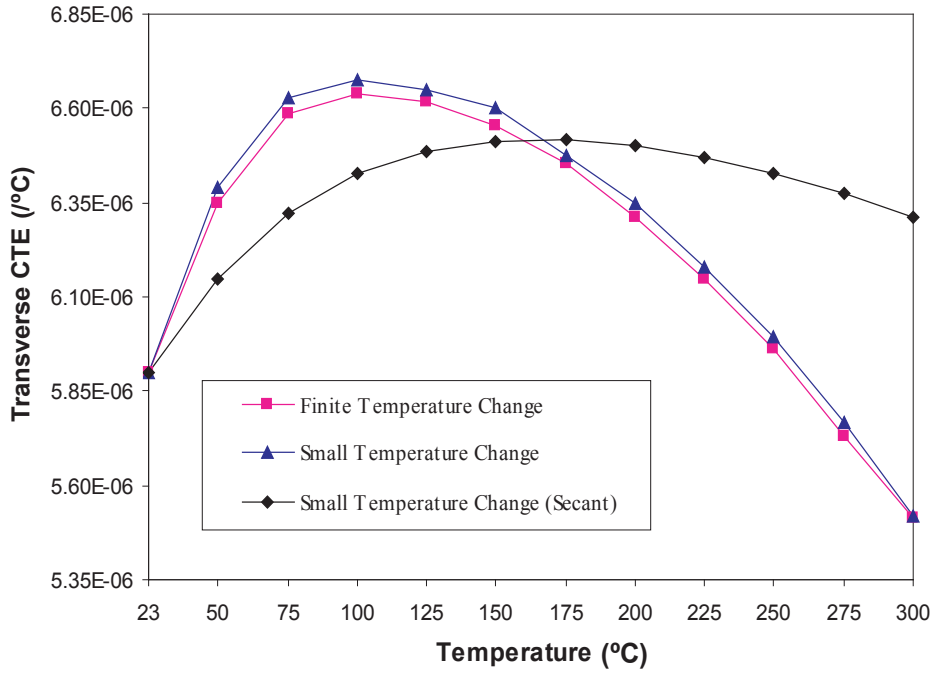

Fig. 3.7: Transverse CTE ($\bar{\alpha}_{22}, \bar{\alpha}_{33}$) change with respect to temperature

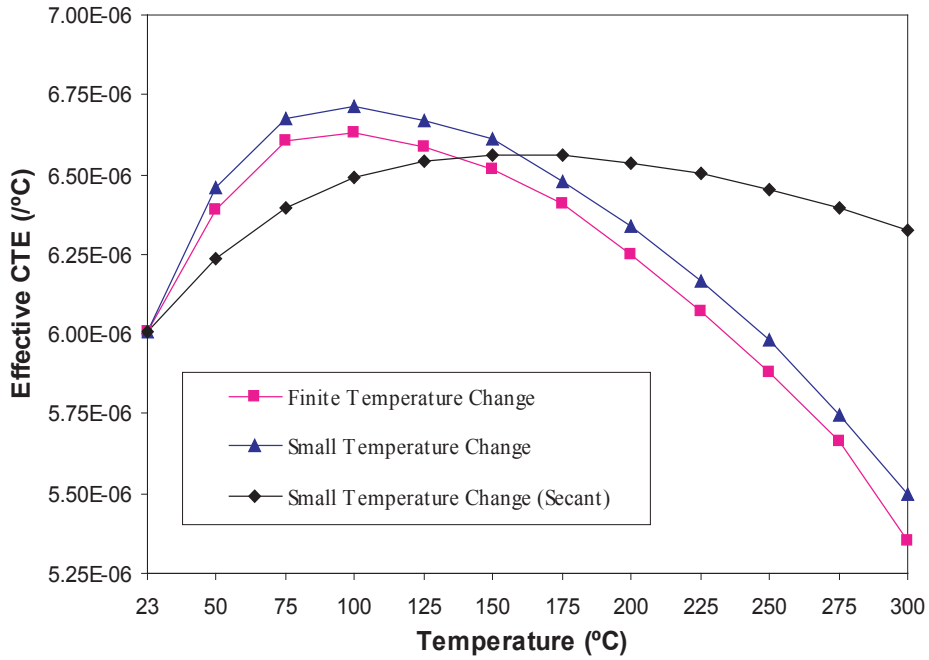

Fig. 3.8: Variation of effective CTE with respect to temperature

VAMUCH to recover the local fields of the fiber reinforced composite with a 0.2 fiber volume fraction. Suppose the material is stress and strain free at room temperature $T_0 = 23°C$. The material is constrained so that there are no displacements and strains. By knowing the fact that constituent 1 has a melting temperature approximately 2700°C and constituent 2 has a melting temperature of 538°C, we slowly increase the temperature of the material from the room temperature all the way to 300°C. Stresses will be generated within the material because of thermal expansion which is constrained by zero deformation. We plot σ_{22} distributions predicted by VAMUCH using both the finite temperature change small strain thermoelasticity and the linear thermoelasticity assuming small temperature changes. The stress field distribution along the lines $y_2 = 0$ and $y_3 = 0$ are plotted in Fig. 3.9 and Fig. 3.10, respectively. We can clearly observe that there are significant differences between the thermal stresses predicted by different theories as the temperature change cannot be considered as small because $(T - T_0)/T_0 = 12.04$. Indeed, we have verified that as we reduce $(T - T_0)/T_0$, the differences between these two predictions decrease.

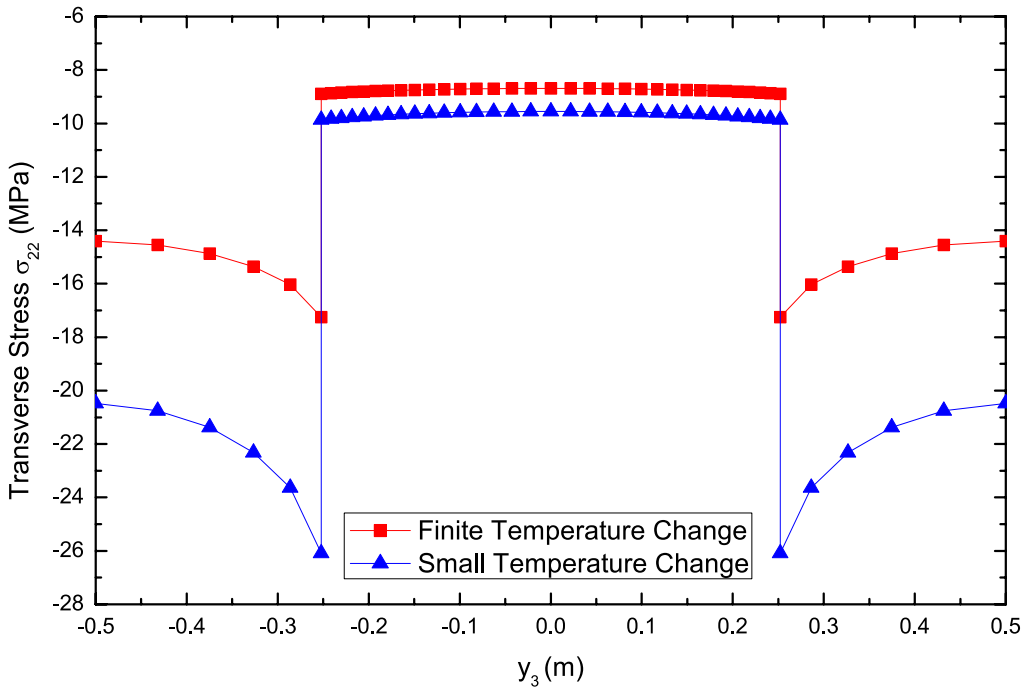

Fig. 3.9: Comparison of transverse stress σ_{22} distribution along $y_2 = 0$

So far, differences between the recovered local stresses predicted by micromechanics models based on traditional linear thermoelasticity and finite temperature change small strain thermoelasticity have been demonstrated in Fig. 3.9 and Fig. 3.10. These differences make the micromechanics model of finite temperature change small strain thermoelasticity becoming important and essential for studying the behaviors of composites under finite temperature change, however, to further verify our new micromechanics model handling finite temperature change, we still need to compare the stress distributions with the finite element results obtained from the structural analysis. To do that, we consider a 3D fiber reinforced composite model with a 0.2 fiber volume fraction. We still propose the assumption that the materials are stress and strain free at room temperature $T_0 = 23°C$. We slowly increase the temperature of this composite from room temperature to 100°C. Instead of fully constraining the materials, we use the exact macroscopic strain field and displacement field as inputs for VAMUCH and the stress distributions are compared with finite element

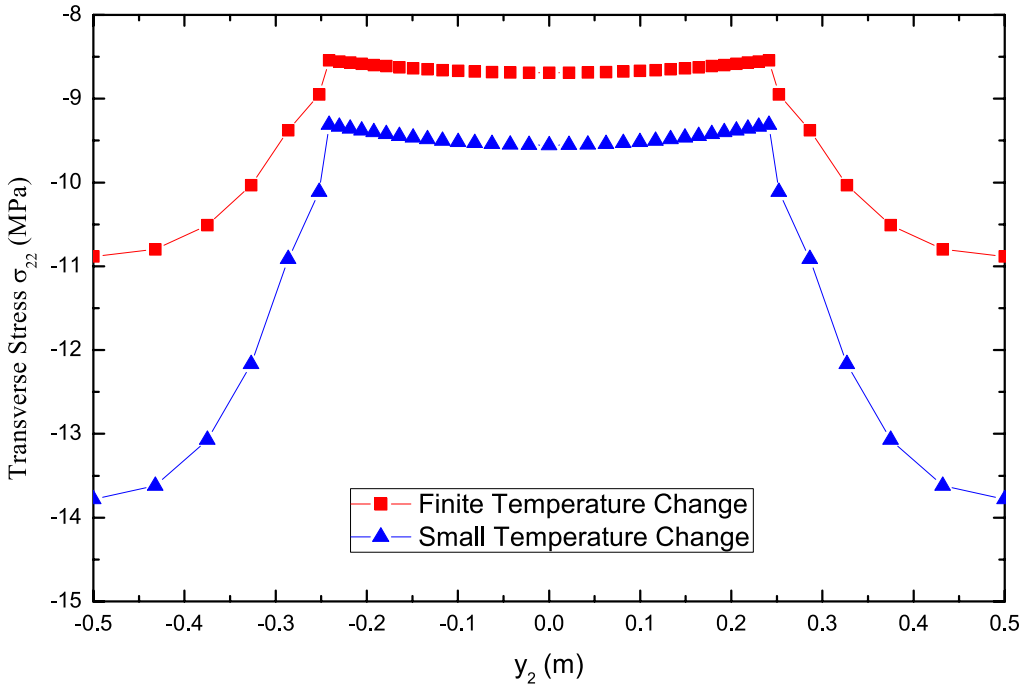

Fig. 3.10: Comparison of transverse stress σ_{22} distribution along $y_3 = 0$

results from ANSYS in order to complete the verification process. We plot σ_{11} distributions along $y_1 = 0$ and $y_2 = 0$ in Figs. 3.11 and 3.12. It can be observed that σ_{11} distributions predicted from VAMUCH and ANSYS are matching each other very well which also proves the newly developed micromechanics model in Chapter 3.

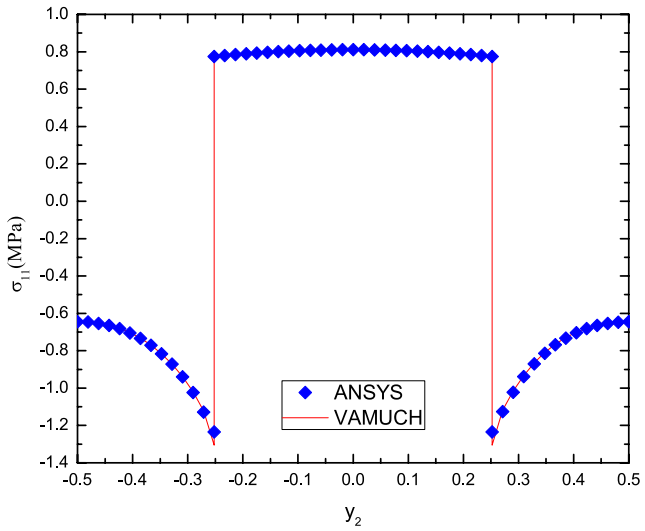

Fig. 3.11: Comparison of σ_{11} distribution along $y_1 = 0$

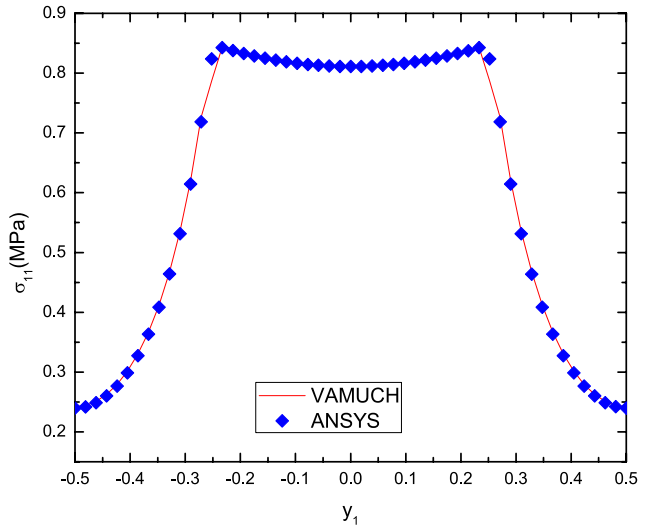

Fig. 3.12: Comparison of σ_{11} distribution along $y_2 = 0$

Chapter 4

Homogenization for Composites with Nonuniformly Distributed Temperature and Loads

In Chapter 3, a thermomechanical micromechanics model is developed using variational asymptotic method in order to deal with temperature dependent constituents and finite temperature change which is capable of dealing with engineering problems with uniform temperature field and no external loads. In this chapter, we are going to extend the work of Chapter 3 and develop another thermomechanical micromechanics model by incorporating the work done by distributed loads into Helmholtz free energy considering nonuniform temperature distribution. Again, the variational asymptotic method is used to formulate the cell problem. Then we implement the cell problem using finite element technique into VAMUCH.

In the previous study, we assume uniform temperature distribution within UC which is normally done in micromechanics modeling of thermomechanical behavior of heterogeneous materials in literature. However, with the current micromechanics model, it is still limited to be used in real engineering systems such as space shuttle thermal protection panels, gas turbine blades, and aerospace heat exchanger fins. Not only finite temperature changes but also nonuniformly distributed temperature fields are experienced in the working conditions of such systems. Also for these systems, they are often experiencing external loads such as pressure in their working conditions. All of these facts reveal us the needs to explicitly incorporate the nonuniformity of temperature field and load effects into the micromechanics model for more accurate predictions.

In this chapter, VAMUCH's capabilities of handling thermomechanical behavior of composites will be enhanced by applying the new developed micromechanics model dealing with nonuniformly distributed temperature and loads. Both the mathematical formulation

and micromechanics model will be demonstrated in the following sections.

4.1 Mathematical Formulation of the Problem

According to Chapter 3, we use the Helmholtz free energy as the functional to govern the thermoelastic behavior of heterogeneous materials, which implies that the nonuniformly distributed temperature field is already known by either designated or thermal conduction or convection analysis. We will not consider the large deformation or finite strain in this case which means even under a large, nonuniform temperature variation, the total strain will still be considered to be small (no creeping, hardening, phase change, or plasticity are considered). This way, we can formulate a simple enough problem to illustrate the effects of nonuniform temperature field and distributed loads to the macroscopic effective properties and local fields. Following derivations in Chapter 3, we can express the Helmholtz free energy f as:

$$f(\varepsilon_{ij}, T) = \frac{1}{2} C_{ijkl}(T)\varepsilon_{ij}\varepsilon_{kl} + l_{ij}(T)\varepsilon_{ij} - \int_{T_0}^{T}\int_{T_0}^{\zeta} \frac{C_{\varepsilon}(0,\rho)}{\rho} \mathrm{d}\rho \mathrm{d}\zeta \tag{4.1}$$

where $C_{ijkl}(T)$ are the temperature dependent fourth-order elasticity tensors, $l_{ij}(T)$ are the temperature dependent second-order thermal stress tensors, $C_{\varepsilon}(0,T)$ is the strain-free heat capacity per unit volume, T_0 is the reference temperature, and T is the current temperature. ε_{ij} are the second-order infinitesimal strain tensors defined as:

$$\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \tag{4.2}$$

where u_i are the displacements, x_i are the Cartesian coordinates describing the heterogeneous material. Usually, the thermal stress tensors $l_{ij}(T)$ are not directly provided as material properties, but rather instantaneous, stress-free coefficients of thermal expansion (CTEs), $\alpha_{ij}(0,T)$ are supplied as material properties. We can compute the secant stress-free CTEs as:

$$\breve{\alpha}_{ij}(T) = \frac{1}{T - T_0}\int_{T_0}^{T} \alpha_{ij}(0,\zeta)\mathrm{d}\zeta \tag{4.3}$$

If small temperature variation is assumed or instantaneous CTEs are constant, secant stress-free CTEs are the same as the instantaneous CTEs. Knowing $\check{\alpha}_{ij}$, we can compute l_{ij} as:

$$l_{ij}(T) = -C_{ijkl}(T)\check{\alpha}_{ij}(T)(T - T_0) = \check{\beta}_{ij}(T)\theta \tag{4.4}$$

Here, for the convenience of programming and to be consistent with the notation used in Chapter 3, we introduced θ as an arbitrary nonzero constant. In other words, we set $\check{\beta}_{ij}(T) = \frac{-C_{ijkl}(T)\check{\alpha}_{ij}(T)(T-T_0)}{\theta}$. Also adding the work done due to applied loads, the thermomechanical behavior of heterogeneous material is obtained by minimizing the following functional:

$$I(u_i) = \int_\Omega (f - b_i u_i)\mathrm{d}\Omega - \int_\Gamma t_i u_i \mathrm{d}\Gamma \tag{4.5}$$

where b_i denote the body forces, t_i denote traction forces applied on the boundary surfaces Γ, and Ω is the domain occupied by the heterogeneous material.

4.2 Micromechanics Model of the Problem

We introduce two cartesian coordinate systems, x_i as the global coordinates to describe the macroscopic behavior of materials and y_i as the local coordinates to describe the microscopic behavior of materials. The relation $y_i = \frac{x_i}{e}$ is used and e is the ratio between two scales that controls size of the unit cell. The displacement fields u_i can be expressed as:

$$u_i(x_1, x_2, x_3, y_1, y_2, y_3) = v_i(x_1, x_2, x_3) + e\psi_i(x_1, x_2, x_3, y_1, y_2, y_3) \tag{4.6}$$

where v_i are components of the global displacement field and $e\psi_i$ are components denoting the differences between u_i and v_i, the so-called fluctuation functions in the literature. Note $e\psi$ enters the formulation together. Although e can be chosen as an arbitrary number of the order of the unit cell size, choosing a different e, the solution will give a different ψ. The value of $e\psi$ remains the same. For this reason, we treat e as a book keeping parameter in derivation and in actual calculations, we set e to be 1 instead.

From Eq. (4.6), it is very easy to understand that the displacement field is changing not only globally but also locally such that the displacement gradient can be evaluated as:

$$\frac{\partial u_i}{\partial x_j} = \frac{1}{e}\frac{\partial u_i}{\partial y_j} + \partial_j u_i \tag{4.7}$$

where $\partial_j u_i$ are the partial derivatives of u_i when y_i are kept constant. The fluctuation functions in Eq. (4.6) can be obtained by minimizing the following functional:

$$\begin{aligned} I_\Omega(\psi_i) = &\left\langle \frac{1}{2} C_{ijkl}(T)\left[\bar{\varepsilon}_{ij} + \psi_{i|j}\right]\left[\bar{\varepsilon}_{kl} + \psi_{k|l}\right] + \check{\beta}_{ij}(T)\left[\bar{\varepsilon}_{ij} + \psi_{i|j}\right]\theta - b_i v_i \right\rangle \\ &- \left\langle \int_{T_0}^{T}\int_{T_0}^{\zeta} \frac{C_\varepsilon(0,\rho)}{\rho} \mathrm{d}\rho \mathrm{d}\zeta \right\rangle - \frac{1}{\Omega}\int_\Gamma t_i(v_i + e\psi_i)\mathrm{d}\Gamma \end{aligned} \tag{4.8}$$

subject to constraints that ψ_i must be periodic with respect to y_i. Here Ω denotes the cell volume,

$$\psi_{i|j} = \frac{1}{2}\left(\frac{\partial \psi_i}{\partial y_j} + \frac{\partial \psi_j}{\partial y_i}\right)$$

$\bar{\varepsilon}_{ij}$ denote the global strain tensors such that:

$$\bar{\varepsilon}_{ij} = \frac{1}{2}\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right)$$

and the angle bracket denotes the average over the unit cell, that is:

$$\langle \cdot \rangle = \frac{1}{\Omega}\int_\Omega (\cdot)\mathrm{d}\Omega$$

Even though this functional in Eq. (4.8) can be solved analytically by applying on a simple case for example a periodically layered composite, we use finite element method to to deal with arbitrary microstructure. We introduce the following matrix notations in VAMUCH:

$$\bar{\varepsilon} = \lfloor \bar{\varepsilon}_{11} \;\; 2\bar{\varepsilon}_{12} \;\; \bar{\varepsilon}_{22} \;\; 2\bar{\varepsilon}_{13} \;\; 2\bar{\varepsilon}_{23} \;\; \bar{\varepsilon}_{33} \rfloor^T \tag{4.9}$$

$$\begin{Bmatrix} \frac{\partial \psi_1}{\partial y_1} \\ \frac{\partial \psi_1}{\partial y_2} + \frac{\partial \psi_2}{\partial y_1} \\ \frac{\partial \psi_2}{\partial y_2} \\ \frac{\partial \psi_1}{\partial y_3} + \frac{\partial \psi_3}{\partial y_1} \\ \frac{\partial \psi_2}{\partial y_3} + \frac{\partial \psi_3}{\partial y_2} \\ \frac{\partial \psi_3}{\partial y_3} \end{Bmatrix} = \begin{bmatrix} \frac{\partial}{\partial y_1} & 0 & 0 \\ \frac{\partial}{\partial y_2} & \frac{\partial}{\partial y_1} & 0 \\ 0 & \frac{\partial}{\partial y_2} & 0 \\ \frac{\partial}{\partial y_3} & 0 & \frac{\partial}{\partial y_1} \\ 0 & \frac{\partial}{\partial y_3} & \frac{\partial}{\partial y_2} \\ 0 & 0 & \frac{\partial}{\partial y_3} \end{bmatrix} \begin{Bmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \end{Bmatrix} \equiv \Gamma_h \psi \tag{4.10}$$

where Γ_h is an operator matrix and ψ is a column matrix containing the three components of the fluctuation functions. If we discretize ψ using the finite element method as:

$$\psi(x_i, y_i) = S(y_i)\mathcal{X}(x_i) \tag{4.11}$$

where S represents the shape function and $\mathcal{X}$ is a column matrix of the nodal values of the fluctuation functions for all active nodes. Substituting Eqs. (4.9), (4.10), and (4.11) into Eq. (4.8), we obtain a discretized version of the functional as:

$$\begin{aligned} I_\Omega =& \frac{1}{2\Omega}\left(\mathcal{X}^T E \mathcal{X} + 2\mathcal{X}^T D_{h\varepsilon}\bar{\varepsilon} + \bar{\varepsilon}^T D_{\varepsilon\varepsilon}\bar{\varepsilon} + 2\mathcal{X}^T D_{h\theta}\theta + 2\mathcal{X}^T D_{hl} + 2\bar{\varepsilon}^T D_{\varepsilon\theta}\theta - 2v^T D_b\right) \\ & - \left\langle \int_{T_0}^{T}\int_{T_0}^{\zeta} \frac{C_\varepsilon(0,\rho)}{\rho} \mathrm{d}\rho \mathrm{d}\zeta \right\rangle \end{aligned} \tag{4.12}$$

where

$$\begin{aligned} E &= \int_\Omega (\Gamma_h S)^T D (\Gamma_h S) \mathrm{d}\Omega \qquad D_{h\varepsilon} = \int_\Omega (\Gamma_h S)^T D \mathrm{d}\Omega \qquad D_{\varepsilon\varepsilon} = \int_\Omega D \ \mathrm{d}\Omega \\ D_{h\theta} &= \int_\Omega (\Gamma_h S)^T \check{\beta} \ \mathrm{d}\Omega \qquad D_{\varepsilon\theta} = \int_\Omega \check{\beta} \ \mathrm{d}\Omega \qquad D_b = \int_\Omega b \ d\Omega + \int_\Gamma t \ \mathrm{d}\Gamma \\ D_{hl} &= -e \int_\Gamma S^T t \ \mathrm{d}\Gamma \end{aligned} \tag{4.13}$$

with D as the 6×6 material matrix condensed from the fourth-order elasticity tensor C_{ijkl}, and $\check{\beta}$ as the 6×1 column condensed from $\check{\beta}_{ij}$. v is a column matrix containing the three components of macroscopic displacement vector, b is a column matrix containing the

three components of the body force vector, and t is a column matrix containing the three components of the traction force vector.

Minimizing I_Ω in Eq. (4.12) with respect to $\mathcal{X}$, we obtain the following linear system:

$$E\mathcal{X} = -D_{h\varepsilon}\bar{\varepsilon} - D_{h\theta}\theta - D_{hl} \tag{4.14}$$

The solution can be written symbolically as:

$$\mathcal{X} = \mathcal{X}_0\bar{\varepsilon} + \mathcal{X}_\theta\theta + \mathcal{X}_l \tag{4.15}$$

Substituting Eq. (4.15) into Eq. (4.12), we can calculate the effective total energy density of the UC as:

$$\bar{I}_\Omega = \frac{1}{2}\bar{\varepsilon}^T\bar{D}\bar{\varepsilon} + \bar{\varepsilon}^T\bar{\beta}\theta + \bar{\varepsilon}^T\bar{l} - v^T\bar{k} + \bar{f}_0 \tag{4.16}$$

with

$$\begin{aligned}
\bar{D} &= \frac{1}{\Omega}(\mathcal{X}_0^T D_{h\varepsilon} + D_{\varepsilon\varepsilon}) \\
\bar{\beta} &= \frac{1}{\Omega}\left(\frac{1}{2}\left(D_{h\varepsilon}^T\mathcal{X}_\theta + \mathcal{X}_0^T D_{h\theta}\right) + D_{\varepsilon\theta}\right) \\
\bar{l} &= \frac{1}{2\Omega}\left(D_{h\varepsilon}^T\mathcal{X}_l + \mathcal{X}_0^T D_{hl}\right) \\
\bar{k} &= \frac{1}{\Omega}D_b \\
\bar{f}_0 &= \frac{1}{2\Omega}\left[\mathcal{X}_l^T D_{hl} + \left(\mathcal{X}_\theta^T D_{hl} + \mathcal{X}_l^T D_{h\theta}\right)\theta + \mathcal{X}_\theta^T D_{h\theta}\theta^2\right] - \left\langle \int_{T_0}^{T}\int_{T_0}^{\zeta}\frac{C_\varepsilon(0,\rho)}{\rho}\mathrm{d}\rho\mathrm{d}\zeta \right\rangle
\end{aligned}$$

where $\bar{D}$ is the effective stiffness matrix, $\bar{\beta}\theta$ is the effective stress tensor induced by temperature change, $\bar{l}$ is the effective stress tensor induced by applied loads, and $\bar{k}$ is the effective body force vector. $\bar{f}_0$ is the effective energy not related with macroscopic strain $\bar{\varepsilon}$ and macroscopic displacement v, caused by temperature change and applied loads, where $\frac{\mathcal{X}_l^T D_{hl}}{2\Omega}$ is purely due to applied loads, $\frac{(\mathcal{X}_\theta^T D_{hl} + \mathcal{X}_l^T D_{h\theta})\theta}{2\Omega}$ is due to the coupling effects of applied loads and temperature, $\frac{\mathcal{X}_\theta^T D_{h\theta}\theta^2}{2\Omega}$ and $\left\langle \int_{T_0}^{T}\int_{T_0}^{\zeta}\frac{C_\epsilon(0,\rho)}{\rho}\mathrm{d}\rho\mathrm{d}\zeta \right\rangle$ are due to temperature change. $\bar{f}_0$ can be used to compute the effective specific heat of the homogenized material.

The effective stress-strain relationship for the homogenized material can be written as:

$$\bar{\sigma} = \bar{D}\bar{\varepsilon} + l^* \tag{4.17}$$

with $l^* = \bar{\beta}\theta + \bar{l}$. Such a stress-strain relationship cannot be directly used in common finite element analysis codes as the effective strain-free stress tensor l^* cannot be used as an input of the material properties. Rather, we can rewrite Eq. (4.17) as:

$$\bar{\sigma} = \bar{D}(\bar{\epsilon} - \alpha^*\theta) \tag{4.18}$$

with $\alpha^* = \frac{-\bar{D}^{-1}l^*}{\theta}$. This constitutive relation along with the effective body force $\bar{k}$ can be used to carry out the macroscopic structural analysis using any standard finite element codes which have the one-way coupled thermoelastic analysis capability. One just needs to let α^* to be the corresponding CTE and θ to be the corresponding temperature change.

If the local fields within the UC are of interest, we can recover those fields after we have obtained the macroscopic behavior which can be described by global displacements v_i and global strains $\bar{\varepsilon}$ [82]:

$$u = v + \begin{bmatrix} \frac{\partial v_1}{\partial x_1} & \frac{\partial v_1}{\partial x_2} & \frac{\partial v_1}{\partial x_3} \\ \frac{\partial v_2}{\partial x_1} & \frac{\partial v_2}{\partial x_2} & \frac{\partial v_2}{\partial x_3} \\ \frac{\partial v_3}{\partial x_1} & \frac{\partial v_3}{\partial x_2} & \frac{\partial v_3}{\partial x_3} \end{bmatrix} \begin{Bmatrix} y_1 \\ y_2 \\ y_3 \end{Bmatrix} + \bar{S}\mathcal{X} \tag{4.19}$$

with u as the column matrix of u_i . Here $\bar{S}$ is different from S due to the recovery of slave nodes and the constrained node. The local strain field can be recovered using:

$$\varepsilon = \bar{\varepsilon} + \Gamma_h \bar{S}\mathcal{X} \tag{4.20}$$

Finally, the local stress field can be recovered straightforwardly using the 3D constitutive relations for the constituent material as:

$$\sigma = D\varepsilon + \check{\beta}\theta = D\varepsilon + l \tag{4.21}$$

4.3 Validation of the Nonuniformly Distributed Temperature and Loads

We consider a periodic binary layered composite with two layers formed with two orthotropic materials and the volume fraction of each layer is taken to be the same. Let ϕ_1 and ϕ_2 denote the volume fractions of first layer and second layer, we have $\phi_1+\phi_2=1$ since we only have two layers for binary composites. It is noticed that we consider the material axes to be the same as the global coordinates y_i so that the material is uniform in y_1-y_2 plane and the periodicity is along y_3 direction. From this point, we can pose the following variational statement of the unit cell:

$$\begin{aligned} J =& \frac{1}{2h}\Bigg[\int_{-\frac{h}{2}}^{(\phi_1-\frac{1}{2})h}\left(\varepsilon^{(1)T}D^{(1)}\varepsilon^{(1)}+2\varepsilon^{(1)T}l^{(1)}-2u^{(1)T}b^{(1)}\right)\mathrm{d}y_3 \\ &+\int_{(\frac{1}{2}-\phi_2)h}^{\frac{h}{2}}\left(\varepsilon^{(2)T}D^{(2)}\varepsilon^{(2)}+2\varepsilon^{(2)T}l^{(2)}-2u^{(2)T}b^{(2)}\right)\mathrm{d}y_3\Bigg]+\lambda_i\left\langle\chi_i\right\rangle \\ &+\beta_{i3}\left[\chi_i^{(2)}(y_3,\frac{h}{2})-\chi_i^{(1)}(y_3,-\frac{h}{2})\right]-u^{(1)T}t^{(1)}-u^{(2)T}t^{(2)}-\left\langle\int_{T_0}^{T}\int_{T_0}^{\zeta}\frac{C_\epsilon(0,\rho)}{\rho}\mathrm{d}\rho\mathrm{d}\zeta\right\rangle \end{aligned} \tag{4.22}$$

with

$$\varepsilon^{(\gamma)}=\lfloor\bar{\varepsilon}_{11}\ \ 2\bar{\varepsilon}_{12}\ \ \bar{\varepsilon}_{22}\ \ 2\bar{\varepsilon}_{13}+\frac{\partial\chi_1^{(\gamma)}}{\partial y_3}\ \ 2\bar{\varepsilon}_{23}+\frac{\partial\chi_2^{(\gamma)}}{\partial y_3}\ \ \bar{\varepsilon}_{33}+\frac{\partial\chi_3^{(\gamma)}}{\partial y_3}\rfloor^T$$

$$l^{(\gamma)}=\lfloor l_{11}^{(\gamma)}\ \ l_{12}^{(\gamma)}\ \ l_{22}^{(\gamma)}\ \ l_{13}^{(\gamma)}\ \ l_{23}^{(\gamma)}\ \ l_{33}^{(\gamma)}\rfloor^T$$

$$u^{(\gamma)}=\lfloor v_1+e\chi_1^{(\gamma)}\ \ v_2+e\chi_2^{(\gamma)}\ \ v_3+e\chi_3^{(\gamma)}\rfloor^T$$

where γ denotes the number of layers and $\chi^{(\gamma)}$ are the fluctuating functions of layers. The material matrices $D^{(\gamma)}$ are characterized by the nine constants for the orthotropic elastic

materials and arranged as:

$$D^{(\gamma)} = \begin{bmatrix} c_{11}^{(\gamma)} & 0 & c_{12}^{(\gamma)} & 0 & 0 & c_{13}^{(\gamma)} \\ 0 & c_{66}^{(\gamma)} & 0 & 0 & 0 & 0 \\ c_{12}^{(\gamma)} & 0 & c_{22}^{(\gamma)} & 0 & 0 & c_{23}^{(\gamma)} \\ 0 & 0 & 0 & c_{55}^{(\gamma)} & 0 & 0 \\ 0 & 0 & 0 & 0 & c_{44}^{(\gamma)} & 0 \\ c_{13}^{(\gamma)} & 0 & c_{23}^{(\gamma)} & 0 & 0 & c_{33}^{(\gamma)} \end{bmatrix} \tag{4.23}$$

In this section, first we verify the new developed model under a nonuniformly distributed temperature field. In order to do so, temperature is predefined as a function of location along y_3 in order to demonstrate the temperature nonuniform distribution. Also we consider the elastic constants of material 1 and 2 as linear functions with respect to location in order to simplify lengthy differential equations inside Mathematica. In this case, we consider the initial temperature T_0 to be 15°C and $T(y_3) = 20 + 10y_3$ as the temperature distribution varies with location and treat e as 1. The material elastic constants are assumed to be $D^{(\gamma)}(y_3) = D_0^{(\gamma)}(1 + \frac{1}{20}y_3)$ and the coefficients of thermal expansion $\alpha^{(\gamma)} = \alpha_0^{(\gamma)}$ are kept the same with different temperatures, but even with this, the thermal stress coefficients are still functions of temperature or location since temperature is a function of location from Eq. (4.4), so we have $\beta^{(\gamma)}(y_3) = -D_0^{(\gamma)}(1 + \frac{1}{20}y_3)\alpha_0^{(\gamma)}$. The needed material property parameters are listed in Table 4.1:

We set up 1D, 2D, and 3D micromechanics models in VAMUCH and compare results with 1D analytical solution carried out by the powerful mathematical software - Mathematica respectively. Due to the fact that we only have limited number of nodes along y_3

Table 4.1: Table of material parameters

γ	$D_0^{(\gamma)}$ (GPa)	$\alpha_0^{(\gamma)}$ (μ/°C)	$\nu^{(\gamma)}$	$\rho^{(\gamma)}$ (kg/m^3)
1	2.76	1.37	0.22	0.001
2	4.0	8.13	0.3	0.001

direction while in Mathematica, a function is applied directly to conduct the final result, we carry out a convergence study in 1D VAMUCH solution by increase the number of nodes along y_3 direction. The corresponding results are listed in Table 4.2:

It is easy to observe that for effective Young's modulus $\bar{E}$ and effective poisson's ratio $\bar{\nu}$ the results from VAMUCH are the same with the analytical solutions provided by Mathematica, and starting from 21 nodes along y_3 direction in VAMUCH, the effective thermal strains $\bar{m}_{11}$, $\bar{m}_{22}$, and $\bar{m}_{33}$ match the analytical solution. For 2D and 3D micromechanics analyses conducted in VAMUCH, we only use a model with two elements, of course, we know if we could have more elements along y_3 direction, the results will be better. However, the convergence study is not duplicated here. Same material properties and temperature distribution are used with 1D VAMUCH solution. The corresponding 2D and 3D results comparing with analytical solution are shown below in Table 4.3 and Table 4.4.

To verify our new micromechanics model proposed in VAMUCH handling external loads and microstructures with voids, we consider a three dimensional finite element solution of a composite with voids formed with 6 elements using Mathematica as shown in Fig. 4.1. The reason that we choose to use FEA results instead of using one dimensional analytical solution to verify our model is that for 1D analytical solution our loads will always be added on the periodic nodes which may bring unnecessary error estimations for calculation. Also in realty, since the microstructures are assumed to repeat many times, loads applied at the edge of unit cell on periodic nodes may not be true. The dimensions of this unit cell

Table 4.2: 1D VAMUCH results compare with Mathematica

	$\bar{E}_1$ ($\bar{E}_2$) (GPa)	$\bar{E}_3$ (GPa)	$\bar{\nu}_{13}$ ($\bar{\nu}_{23}$)	$\bar{\nu}_{12}$	$\bar{m}_{11}$ ($\bar{m}_{22}$) (10^{-5})	$\bar{m}_{33}$ (10^{-5})
Mathematica	3.3933	3.2592	0.2598	0.2687	3.9363	3.1326
VAMUCH (3 pts)	3.3933	3.2593	0.2598	0.2687	3.9305	3.1367
VAMUCH (5 pts)	3.3933	3.2592	0.2598	0.2687	3.9349	3.1336
VAMUCH (11 pts)	3.3933	3.2592	0.2598	0.2687	3.9305	3.1367
VAMUCH (21 pts)	3.3933	3.2592	0.2598	0.2687	3.9362	3.1326
VAMUCH (41 pts)	3.3933	3.2592	0.2598	0.2687	3.9363	3.1326
VAMUCH (101 pts)	3.3933	3.2592	0.2598	0.2687	3.9363	3.1326

Table 4.3: 2D VAMUCH results compare with Mathematica

	$\bar{E}_1$ ($\bar{E}_2$) (GPa)	$\bar{E}_3$ (GPa)	$\bar{\nu}_{13}$ ($\bar{\nu}_{23}$)	$\bar{\nu}_{12}$	$\bar{m}_{11}$ ($\bar{m}_{22}$) (10^{-5})	$\bar{m}_{33}$ (10^{-5})
Mathematica	3.3933	3.2592	0.2598	0.2687	3.9363	3.1326
2D VAMUCH (2 elems)	3.3933	3.2593	0.2598	0.2687	3.9363	3.1415

Table 4.4: 3D VAMUCH results compare with Mathematica

	$\bar{E}_1$ ($\bar{E}_2$) (GPa)	$\bar{E}_3$ (GPa)	$\bar{\nu}_{13}$ ($\bar{\nu}_{23}$)	$\bar{\nu}_{12}$	$\bar{m}_{11}$ ($\bar{m}_{22}$) (10^{-5})	$\bar{m}_{33}$ (10^{-5})
Mathematica	3.3933	3.2592	0.2598	0.2687	3.9363	3.1326
3D VAMUCH (2 elems)	3.3933	3.2593	0.2598	0.2687	3.9363	3.1415

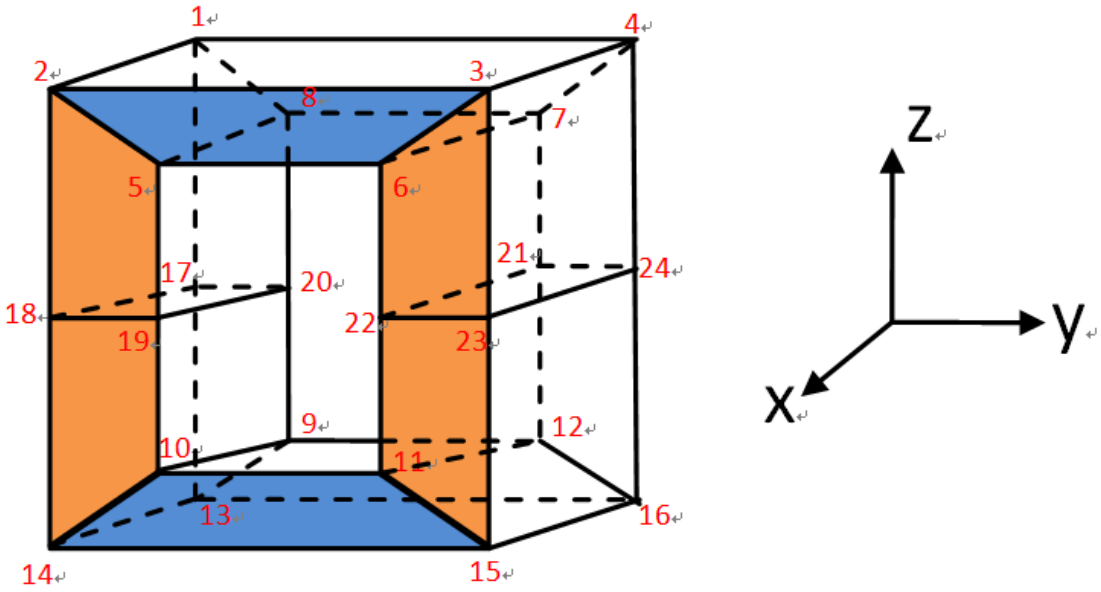

Fig. 4.1: The diagrammatic sketch of a six element composite with voids

are $y_1 \in \{-0.5, 0.5\}$, $y_2 \in \{-0.5, 0.5\}$, and $y_3 \in \{-0.5, 0.5\}$. The material properties of constituent 1 (formed by top and bottom two elements) and constituent 2 (formed by left and right four middle elements) are directly taken from Table 4.1. The periodic pressure loads of 6MPa applied on all six inner surfaces. The resulting effective properties of this microstructure as we predicted are exactly the same with VAMUCH results, so only one set of results will be provided in the following. The results of effective elastic constant $\bar{D}$, effective stress due to applied loads $\bar{l}$, effective strain due to applied loads $\bar{m} = -\bar{D}^{-1}\bar{l}$, and effective body force $\bar{k}$ are listed below in matrices (4.24), (4.24), (4.24), and (4.24).

The effective stiffness matrix $\bar{D}$ (GPa):

$$\bar{D} = \begin{bmatrix} 2.47847 & 0 & 0.474283 & 0 & 0 & 0.71275 \\ 0 & 0.581918 & 0 & 0 & 0 & 0 \\ 0.474283 & 0 & 1.52521 & 0 & 0 & 0.394552 \\ 0 & 0 & 0 & 0.769221 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.596054 & 0 \\ 0.71275 & 0 & 0.394552 & 0 & 0 & 2.19693 \end{bmatrix}$$

The effective stress $\bar{l}$ (MPa) due to applied loads:

$$\bar{l} = \begin{bmatrix} -0.710935 \\ 0 \\ -1.44656 \\ 0 \\ 0 \\ -1.24835 \end{bmatrix}$$

The effective strain $\bar{m}$ due to applied loads:

$$\bar{m} = \begin{bmatrix} 6.87917 \times 10^{-6} \\ 0 \\ 8.38847 \times 10^{-4} \\ 0 \\ 0 \\ 4.15341 \times 10^{-4} \end{bmatrix}$$

The effective body force $\bar{k}$ (MPa):

$$\bar{k} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Second, the stress and strain field are recovered and compared with FEA results from Mathematica by assuming the unit cell has a macroscopic strain field $\bar{\epsilon}_{33} = 0.0001$ after structural analysis. The recovered nodal displacements u_i, strains ϵ_{ij}, and stresses σ_{ij} within each elements are provided below and as we predicted two set of results are matching each other. Again, we only provide one set of solutions for space saving.

Here only u_2 and u_3 are listed in Table 4.5 because for this particular macroscopic strain field, the displacement field u_1 for all nodes is zero.

4.4 Numerical Examples

Several numerical examples are used to validate and demonstrate the new capability of handling nonuniformly distributed temperature fields and loads implemented in VAMUCH. As we know, micromechanics models should be able to predict the local fields accurately which is also the ultimate goal of micromechanics modeling. So in this section, we focus on comparing the local stress fields of these examples after micromechanical recovery analyses in VAMUCH with the finite element results provided by ANSYS. The validation process is

Table 4.5: The local displacement fields for each node

Node Number	u_2	u_3
1	0	0.00005
2	0	0.00005
3	0	0.00005
4	0	0.00005
5	-0.000152974	0.000264354
6	0.000152974	0.000264354
7	0.000152974	0.000264354
8	-0.000152974	0.000264354
9	-0.000152974	-0.000264354
10	-0.000152974	-0.000264354
11	0.000152974	-0.000264354
12	0.000152974	-0.000264354
13	0	-0.00005
14	0	-0.00005
15	0	-0.00005
16	0	-0.00005
17	0	0
18	0	0
19	-0.000235723	0
20	-0.000235723	0
21	0.000235723	0
22	0.000235723	0
23	0	0
24	0	0

Table 4.6: The local strain fields for element 1

Node Number	ϵ_{11}	$2\epsilon_{12}$	ϵ_{22}	$2\epsilon_{13}$	$2\epsilon_{23}$	ϵ_{33}
8	0	0	0.000455666	0	0.00122053	-0.00107177
5	0	0	0.000455666	0	0.00122053	-0.00107177
6	0	0	0.000455666	0	-0.00122053	-0.00107177
7	0	0	0.000455666	0	-0.00122053	-0.00107177
1	0	0	-0.0000325476	0	0.00073232	-0.00107177
2	0	0	-0.0000325476	0	0.00073232	-0.00107177
3	0	0	-0.0000325476	0	-0.00073232	-0.00107177
4	0	0	-0.0000325476	0	-0.00073232	-0.00107177

Table 4.7: The local stress fields for element 1

Node Number	σ_{11} (MPa)	σ_{12} (MPa)	σ_{22} (MPa)	σ_{13} (MPa)	σ_{23} (MPa)	σ_{33} (MPa)
8	-0.547568	0	0.483283	0	1.3806	-2.97223
5	-0.547568	0	0.483283	0	1.3806	-2.97223
6	-0.547568	0	0.483283	0	-1.3806	-2.97223
7	-0.547568	0	0.483283	0	-1.3806	-2.97223
1	-0.981472	0	-1.0551	0	0.828362	-3.40613
2	-0.981472	0	-1.0551	0	0.828362	-3.40613
3	-0.981472	0	-1.0551	0	-0.828362	-3.40613
4	-0.981472	0	-1.0551	0	-0.828362	-3.40613

Table 4.8: The local strain fields for element 2

Node Number	ϵ_{11}	$2\epsilon_{12}$	ϵ_{22}	$2\epsilon_{13}$	$2\epsilon_{23}$	ϵ_{33}
17	0	0	-0.00117861	0	-0.0000176062	0.0000501375
18	0	0	-0.00117861	0	-0.0000176062	0.0000501375
19	0	0	-0.00117861	0	0.000246487	0.000798076
20	0	0	-0.00117861	0	0.000246487	0.000798076
1	0	0	-0.000782474	0	0.0011043	0.0000501375
2	0	0	-0.000782474	0	0.0011043	0.0000501375
5	0	0	-0.00051838	0	0.00211633	0.000798076
8	0	0	-0.00051838	0	0.00211633	0.000798076

Table 4.9: The local stress fields for element 2

Node Number	σ_{11} (MPa)	σ_{12} (MPa)	σ_{22} (MPa)	σ_{13} (MPa)	σ_{23} (MPa)	σ_{33} (MPa)
17	-2.60418	0	-6.23068	0	-0.0270865	-2.44991
18	-2.60418	0	-6.23068	0	-0.0270865	-2.44991
19	-0.878166	0	-4.50467	0	0.379211	1.57745
20	-0.878166	0	-4.50467	0	0.379211	1.57745
1	-1.69001	0	-4.09762	0	1.69892	-1.53574
2	-1.69001	0	-4.09762	0	1.69892	-1.53574
5	0.64545	0	-0.949566	0	3.2559	3.10107
8	0.64545	0	-0.949566	0	3.2559	3.10107

Table 4.10: The local strain fields for element 3

Node Number	ϵ_{11}	$2\epsilon_{12}$	ϵ_{22}	$2\epsilon_{13}$	$2\epsilon_{23}$	ϵ_{33}
21	0	0	-0.00117861	0	-0.000246487	0.000798076
22	0	0	-0.00117861	0	-0.000246487	0.000798076
23	0	0	-0.00117861	0	0.0000176062	0.0000501375
24	0	0	-0.00117861	0	0.0000176062	0.0000501375
7	0	0	-0.00051838	0	-0.00211633	0.000798076
6	0	0	-0.00051838	0	-0.00211633	0.000798076
3	0	0	-0.000782474	0	-0.0011043	0.0000501375
4	0	0	-0.000782474	0	-0.0011043	0.0000501375

Table 4.11: The local stress fields for element 3

Node Number	σ_{11} (MPa)	σ_{12} (MPa)	σ_{22} (MPa)	σ_{13} (MPa)	σ_{23} (MPa)	σ_{33} (MPa)
21	-0.878166	0	-4.50467	0	-0.379211	1.57745
22	-0.878166	0	-4.50467	0	-0.379211	1.57745
23	-2.60418	0	-6.23068	0	0.0270865	-2.44991
24	-2.60418	0	-6.23068	0	0.0270865	-2.44991
7	0.64545	0	-0.949566	0	-3.2559	3.10107
6	0.64545	0	-0.949566	0	-3.2559	3.10107
3	-1.69001	0	-4.09762	0	-1.69892	-1.53574
4	-1.69001	0	-4.09762	0	-1.69892	-1.53574

Table 4.12: The local strain fields for element 4

Node Number	ϵ_{11}	$2\epsilon_{12}$	ϵ_{22}	$2\epsilon_{13}$	$2\epsilon_{23}$	ϵ_{33}
13	0	0	-0.0000325476	0	-0.00073232	-0.00107177
14	0	0	-0.0000325476	0	-0.00073232	-0.00107177
15	0	0	-0.0000325476	0	0.00073232	-0.00107177
16	0	0	-0.0000325476	0	0.00073232	-0.00107177
9	0	0	0.000455666	0	-0.00122053	-0.00107177
10	0	0	0.000455666	0	-0.00122053	-0.00107177
11	0	0	0.000455666	0	0.00122053	-0.00107177
12	0	0	0.000455666	0	0.00122053	-0.00107177

Table 4.13: The local stress fields for element 4

Node Number	σ_{11} (MPa)	σ_{12} (MPa)	σ_{22} (MPa)	σ_{13} (MPa)	σ_{23} (MPa)	σ_{33} (MPa)
13	-0.981472	0	-1.0551	0	-0.828362	-3.40613
14	-0.981472	0	-1.0551	0	-0.828362	-3.40613
15	-0.981472	0	-1.0551	0	0.828362	-3.40613
16	-0.981472	0	-1.0551	0	0.828362	-3.40613
9	-0.547568	0	0.483283	0	-1.3806	-2.97223
10	-0.547568	0	0.483283	0	-1.3806	-2.97223
11	-0.547568	0	0.483283	0	1.3806	-2.97223
12	-0.547568	0	0.483283	0	1.3806	-2.97223

Table 4.14: The local strain fields for element 5

Node Number	ϵ_{11}	$2\epsilon_{12}$	ϵ_{22}	$2\epsilon_{13}$	$2\epsilon_{23}$	ϵ_{33}
13	0	0	-0.000782474	0	-0.0011043	0.0000501375
14	0	0	-0.000782474	0	-0.0011043	0.0000501375
10	0	0	-0.00051838	0	-0.00211633	0.000798076
9	0	0	-0.00051838	0	-0.00211633	0.000798076
17	0	0	-0.00117861	0	0.0000176062	0.0000501375
18	0	0	-0.00117861	0	0.0000176062	0.0000501375
19	0	0	-0.00117861	0	-0.000246487	0.000798076
20	0	0	-0.00117861	0	-0.000246487	0.000798076

Table 4.15: The local stress fields for element 5

Node Number	σ_{11} (MPa)	σ_{12} (MPa)	σ_{22} (MPa)	σ_{13} (MPa)	σ_{23} (MPa)	σ_{33} (MPa)
13	-1.69001	0	-4.09762	0	-1.69892	-1.53574
14	-1.69001	0	-4.09762	0	-1.69892	-1.53574
10	0.64545	0	-0.949566	0	-3.2559	3.10107
9	0.64545	0	-0.949566	0	-3.2559	3.10107
17	-2.60418	0	-6.23068	0	0.0270865	-2.44991
18	-2.60418	0	-6.23068	0	0.0270865	-2.44991
19	-0.878166	0	-4.50467	0	-0.379211	1.57745
20	-0.878166	0	-4.50467	0	-0.379211	1.57745

Table 4.16: The local strain fields for element 6

Node Number	ϵ_{11}	$2\epsilon_{12}$	ϵ_{22}	$2\epsilon_{13}$	$2\epsilon_{23}$	ϵ_{33}
12	0	0	-0.00051838	0	0.00211633	0.000798076
11	0	0	-0.00051838	0	0.00211633	0.000798076
15	0	0	-0.000782474	0	0.0011043	0.0000501375
16	0	0	-0.000782474	0	0.0011043	0.0000501375
21	0	0	-0.00117861	0	0.000246487	0.000798076
22	0	0	-0.00117861	0	0.000246487	0.000798076
23	0	0	-0.00117861	0	-0.0000176062	0.0000501375
24	0	0	-0.00117861	0	-0.0000176062	0.0000501375

Table 4.17: The local stress fields for element 6

Node Number	σ_{11} (MPa)	σ_{12} (MPa)	σ_{22} (MPa)	σ_{13} (MPa)	σ_{23} (MPa)	σ_{33} (MPa)
12	0.64545	0	-0.949566	0	3.2559	3.10107
11	0.64545	0	-0.949566	0	3.2559	3.10107
15	-1.69001	0	-4.09762	0	1.69892	-1.53574
16	-1.69001	0	-4.09762	0	1.69892	-1.53574
21	-0.878166	0	-4.50467	0	0.379211	1.57745
22	-0.878166	0	-4.50467	0	0.379211	1.57745
23	-2.60418	0	-6.23068	0	-0.0270865	-2.44991
24	-2.60418	0	-6.23068	0	-0.0270865	-2.44991

based on a step by step procedure, in other words, we first validate the new capability of handling nonuniformly distributed temperature field, then we validate the new capability of handling load effects, finally we validate the new capability of handling both nonuniformly temperature field and loads.

4.4.1 Two-phase Composites under Nonuniform Temperature Field

We consider a fulfilled two-phase composite with the microstructure as shown in Fig. 4.2. The dimensionality of this unit cell is 1m×1m×1m. Inconel 625 is used as the constituent material 1 as shown in lighter color and Acier inox Z10 is used as the constituent material 2 as shown in darker color in Fig. 4.2. The constituent materials are considered as isotropic with temperature dependent material properties including Young's modulus $E(T)$, Poisson's ratio $\nu(T)$, coefficient of thermal expansion $\alpha(T)$, density $\rho(T)$, and thermal conductivity $K(T)$, given in Table 4.18 and Table 4.19. Moveover, both materials are assumed stress and strain free at 20°C.

In order to mimic the nonuniformly distributed temperature within the microstructure in real practical problems, we set up heat conduction analysis first. We assume that this unit cell has a higher temperature field of 400°C on top and bottom surfaces, and a lower temperature field of 20°C on left and right surfaces. After a heat conduction analysis, the resulting nonuniformly distributed temperature field is obtained as shown in Fig. 4.3. A one-way thermomechanical coupling is used to study the behavior of this microstructure under this known nonuniformly distributed temperature field in both VAMUCH and ANSYS. To conduct the recovery analysis in VAMUCH, we use the exact macroscopic displacements and displacement gradients from the structural analysis as our inputs. The macroscopic displacement fields and their gradients can be obtained by plugging the homogenized unit cell with effective material properties back into the original model to form a homogenized material and run the structural analysis with it. The stress will be generated within the unit cell because of the thermal expansion under this given nonuniformly distributed temperature field. We plot σ_{11}, σ_{22}, and σ_{33} along y_1 direction at the location where $y_2 = 0.125$m and $y_3 = 0.5$m. The stress distributions from both VAMUCH and ANSYS are plotted in

Figs. 4.4, 4.5, and 4.6. Excellent matches between these two approaches can be clearly observed from the plots which demonstrates the capability and accuracy of VAMUCH for predicting the local fields of heterogeneous materials with nonuniformly distributed temperature fields.

4.4.2 Two-phase Composites with Voids under Nonuniform Temperature Field

In the last numerical example, the capability of VAMUCH handling fulfilled two-phase composites has been investigated and proved. However, in real practical applications, such fulfilled composites sometimes are not as often used for porous composites for which the thermomechanical behavior is important. For this reason, a further investigation and validation of the new developed theory handling microstructure with voids under nonuniformly distributed temperature field is needed.

We consider a two-phase composite with voids as shown in Fig. 4.7. We still use Inconel 625 as constituent material 1 and Acier inox Z10 as constituent material 2, and assume these two materials are stress and strain free at 20°C. Suppose we have a heat supply which gives a temperature field of 100°C on the inner surfaces of the frame and the temperature of upper and bottom surfaces of the UC is kept at 20°C. In this case, the heat conduction analysis can be carried out and the resulting temperature field of the microstructure is shown in Fig. 4.8. Using this nonuniformly distributed temperature field and the real macroscopic fields obtained from structural analysis, we can recover the local stress fields of this microstructure. We plot σ_{11}, σ_{22}, and σ_{33} along y_2 direction at the location where $y_1 = 0.275$m and $y_3 = 0.5$m. This specific location basically starts from top surface going through one of the frame of the structure and reaches the bottom surface. Again, perfect matches between VAMUCH and ANSYS results are observed as plotted in Figs. 4.9, 4.10, and 4.11.

4.4.3 Aerospace Heat exchanger Fins under Nonuniform Temperature Field

The above examples have demonstrated the capability of VAMUCH handling both fulfilled and voided microstructures under nonuniformly distributed temperature fields. How-

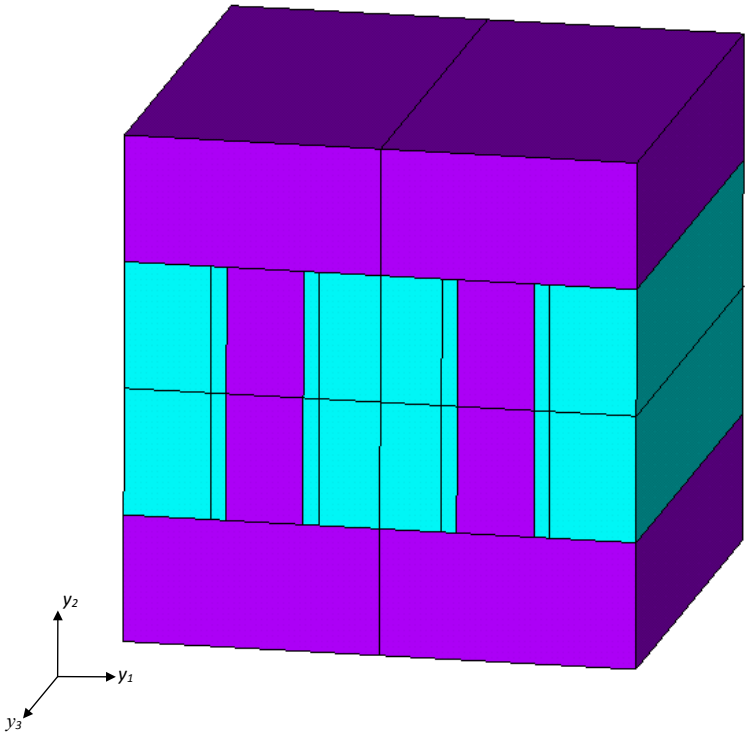

Fig. 4.2: Microstructure of a fulfilled two-phase composite

Table 4.18: Temperature dependent material properties for Inconel 625

T (°C)	E_1 (GPa)	ν_1	α_1 (μ/°C)	ρ_1 (kg/m^3)	K_1 (W/m-°C)
20	205.441	0.278	12.78	8440	9.8
100	190.342	0.280	12.79	8440	9.8
150	184.835	0.282	12.87	8440	9.8
200	181.137	0.284	12.95	8440	9.8
250	178.281	0.286	13.11	8440	9.8
300	175.507	0.288	13.27	8440	9.8
350	173.689	0.291	13.43	8440	9.8
400	172.302	0.294	13.59	8440	9.8
450	171.131	0.298	13.76	8440	9.8
500	170.206	0.302	13.92	8440	9.8

Table 4.19: Temperature dependent material properties for Acier inox Z10

T (°C)	E_2 (GPa)	ν_2	α_2 (μ/°C)	ρ_2 (kg/m^3)	K_2 (W/m-°C)
20	199.948	0.290	15.47	7900	15
100	195.469	0.290	16.26	7900	15
150	191.870	0.290	16.70	7900	15
200	188.271	0.290	17.15	7900	15
250	184.672	0.290	17.37	7900	15
300	181.073	0.290	17.58	7900	15
350	177.474	0.290	17.84	7900	15
400	173.875	0.290	18.12	7900	15
450	170.276	0.290	18.36	7900	15
500	166.677	0.290	18.57	7900	15

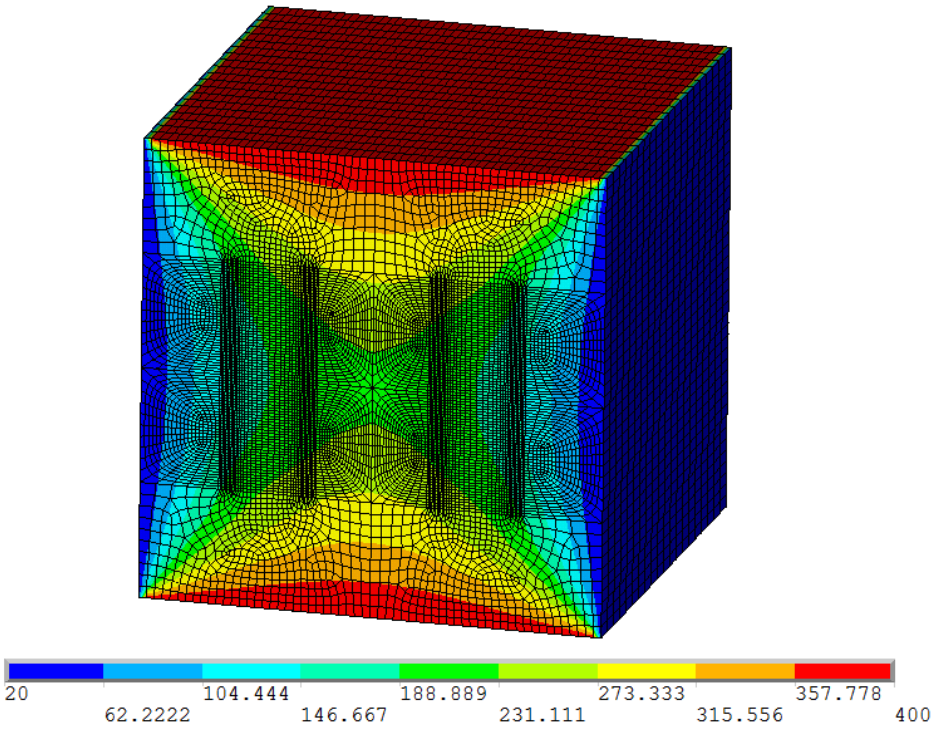

Fig. 4.3: Temperature distribution of UC after a heat conduction analysis

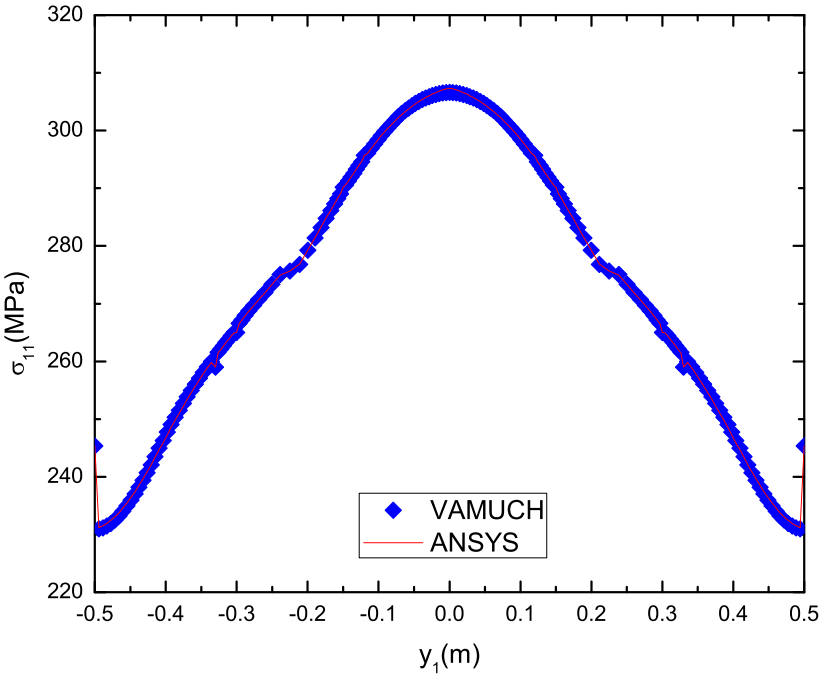

Fig. 4.4: Comparison of thermal stress σ_{11} along y_1 at $y_2 = 0.125$m and $y_3 = 0.5$m

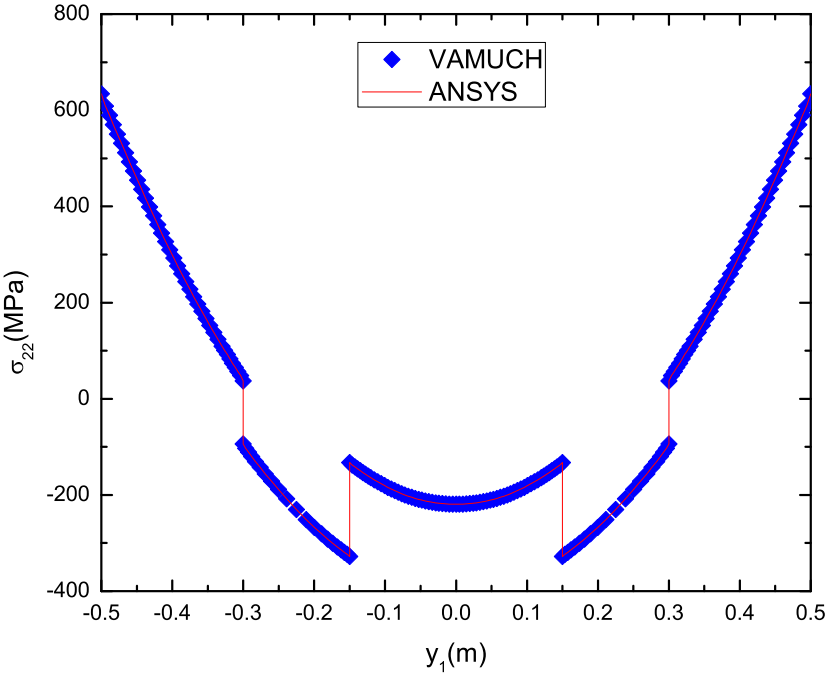

Fig. 4.5: Comparison of thermal stress σ_{22} along y_1 at $y_2 = 0.125$m and $y_3 = 0.5$m

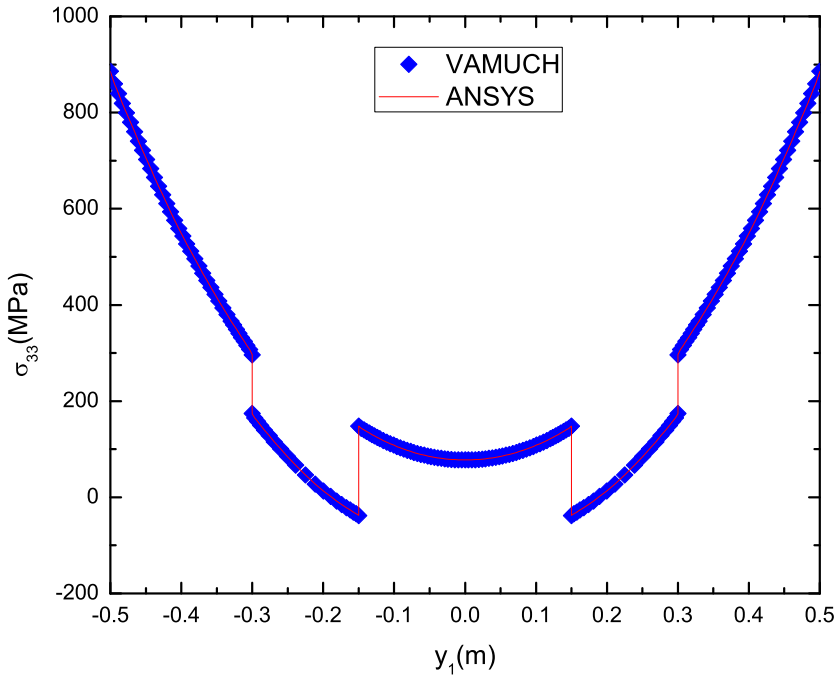

Fig. 4.6: Comparison of thermal stress σ_{33} along y_1 at $y_2 = 0.125$m and $y_3 = 0.5$m

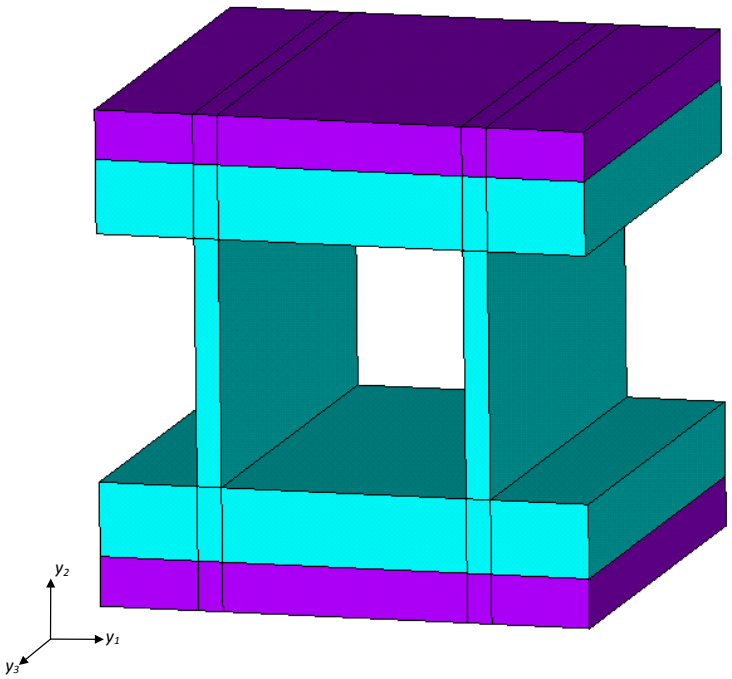

Fig. 4.7: Microstructure of a two-phase composite with voids

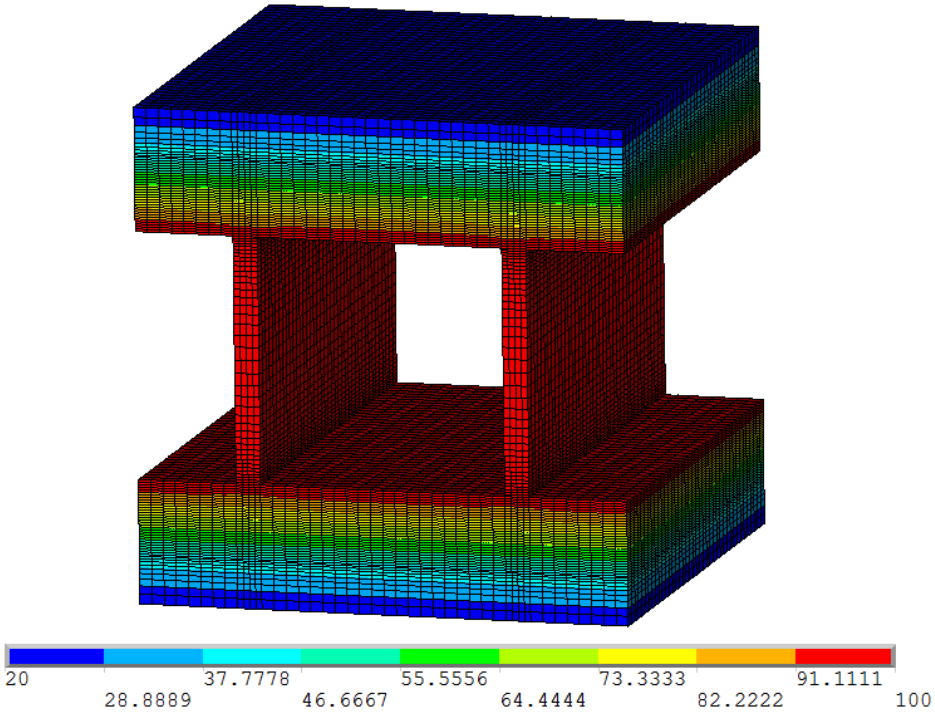

Fig. 4.8: Temperature distribution of UC after a heat conduction analysis

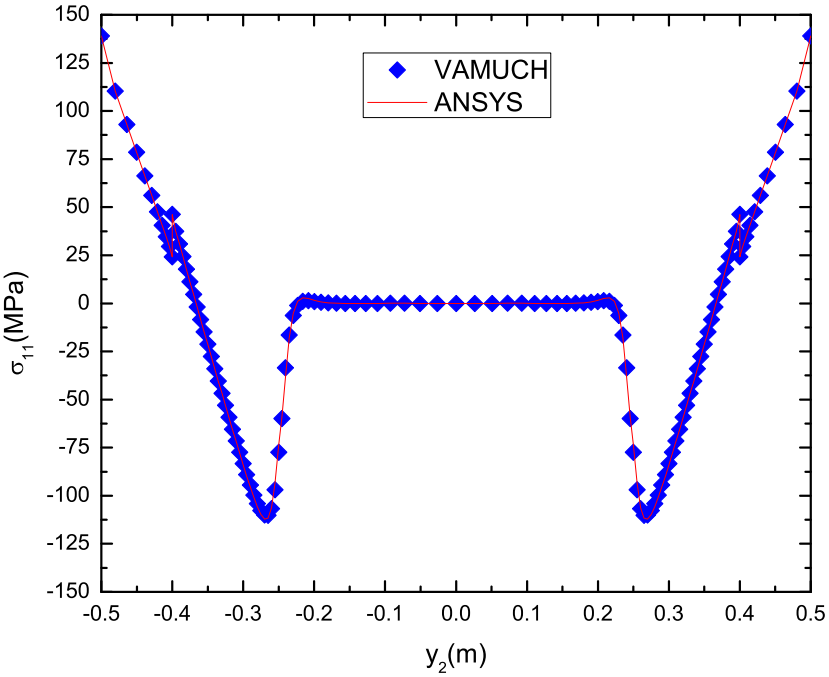

Fig. 4.9: Comparison of thermal stress σ_{11} along y_2 at $y_1 = 0.275$m and $y_3 = 0.5$m

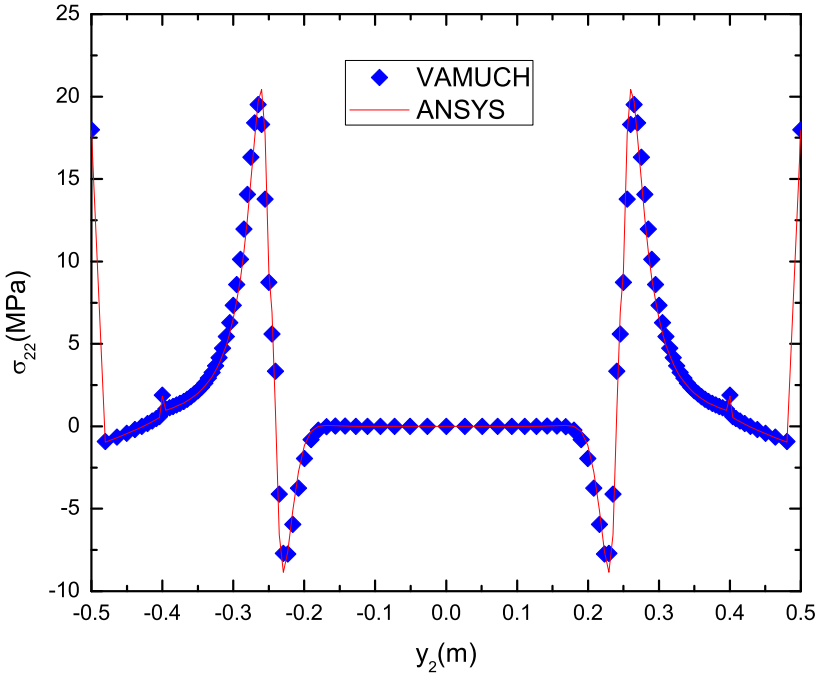

Fig. 4.10: Comparison of thermal stress σ_{22} along y_2 at $y_1 = 0.275$m and $y_3 = 0.5$m

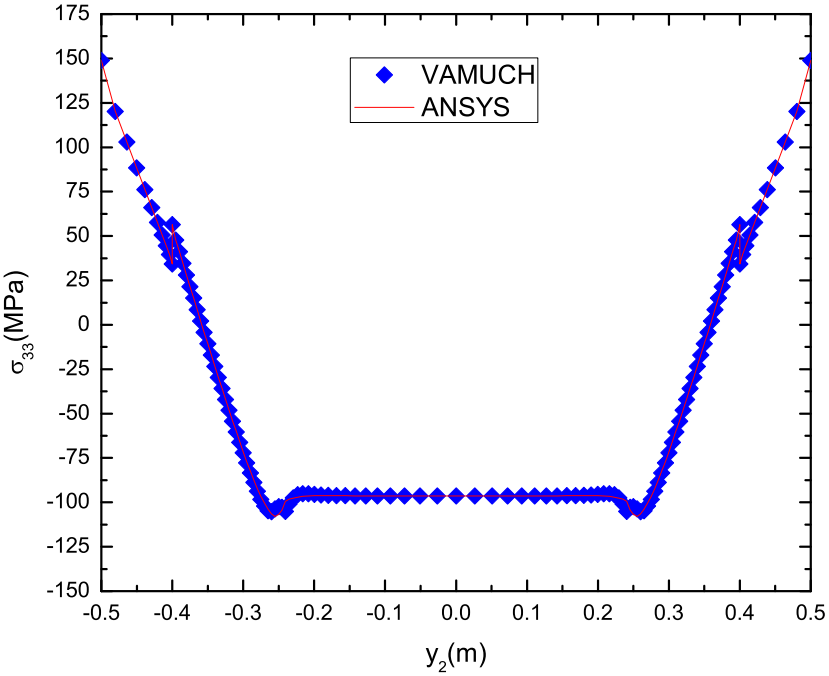

Fig. 4.11: Comparison of thermal stress σ_{33} along y_2 at $y_1 = 0.275$m and $y_3 = 0.5$m

ever, it is more challenging to validate the results with realistic engineering microstructures as these problems have not been studied using micromechanics in the literature. But the above examples are needed as it is a rigorous validation procedure carried in our study. In this example, we conduct a micromechanical analysis of a unit cell of aerospace heat exchanger fins which possesses very thin frame structures (aka highly porous materials) and experiences large temperature changes with nonuniformly distributed temperature fields induced by hot and cold airflows under their working conditions. The unit cell is shown in Fig. 4.12 with a dimensionality of 3.62mm×7.7272mm×2.58mm. The upper part of this unit cell is taken from the layer that hot airflow going through (aka hot cell) while the lower part is taken from the layer that cold airflow going through (aka cold cell). As we can see, these two layers are usually stacked with a 90° lay up angle in order to achieve a better temperature exchange. The double cell is formed with three different materials: the parting sheets are made by Inconel 625 as in Table 4.18, the hot fins are made by Acier inox Z10 as in Table 4.19, and the cold fins are made by Nickel 201 as in Table 4.20. Under the nonuniformly distributed temperature field as shown in Fig. 4.12 and the real macroscopic fields obtained from structural analysis, we can recover the local stress fields of this microstructure and compare with ANSYS results. We plot σ_{11}, σ_{22}, and σ_{33} along y_2 direction at the location where $y_1 = 0.905$mm and $y_3 = 0.645$mm as indicated in Fig. 4.13. The stress results are plotted in Figs. 4.14, 4.15, and 4.16. The stress results are matched each other for σ_{11} and σ_{33} while for σ_{22} they are not matched very well at the parting sheet regions. After several testings, we figured out that the difference is caused by the fairly small thickness of parting sheets with which even if we apply a uniform temperature field on this unit cell, difference will still be there at those regions. Of course, if we increase the element numbers at those regions will improve the results but so far with the limited computational ability, these results presented in Figs. 4.14, 4.15, and 4.16 are the best ones that we can get. In other words, if we increase the thickness of those regions, the unit cell will be similar like stacking two of the two-phase composites models with voids in the second example together with a 90° lay up angle, we have validated that VAMUCH σ_{22} results agree with ANSYS

very well for that case. It is also worth to notice that ANSYS result of σ_{22} for this heat exchanger fin unit cell case, at the regions close to top and bottom parting sheets, does not perform a very smooth transition of stress as shown in Fig 4.15 while VAMUCH result still perform a smooth transition. For the sense of this microstructure and corresponding temperature fields, we believe VAMUCH result of σ_{22} is more trustful in this situation.

4.4.4 Two-phase Composites with Voids under Pressure Loads

Next, we are interested to see the comparison of recovered stress fields between VAMUCH and ANSYS with external loads applied on the microstructure. So we consider the unit cell experiences a consecutive flow of hot air from one end of the middle frame to the other end in y_3 direction. This heat source keeps the unit cell at a temperature of 400°C and gives pressure loads of 0.6MPa on all inner surfaces as shown in Fig. 4.17. We still assume the materials are stress and strain free at 20°C. Constituent material 1 is Inconel 625 and material 2 is Acier inox Z10. Use the macroscopic displacement fields and strain fields obtained from structural analysis, we can recover the local stresses and compare with ANSYS results. The stress results are plotted along y_2 direction at the location where $y_1 = 0.225$m and $y_3 = 0.25$m in Figs. 4.18, 4.19, and 4.20. We found very good matches between VAMUCH results and ANSYS results, which validates our new micromechanics model handling external loads.

4.4.5 A Hot Cell under Pressure Loads

We consider a hot cell (3.62mm×3.1416mm×2.58mm) with uniform pressure loads of 0.6MPa as shown in Fig. 4.22 applied on the inner frame surfaces. This pressure load is based on the real working condition of hot cell. Since the elements are too dense in Fig. 4.22, we provide another geometric picture of hot cell as shown in Fig. 4.21 in order to have a better view. The temperature field of hot cell is kept at 450°C and we assume the materials are stress and strain free at 20°C. The parting sheets are made by Inconel 625 as in Table 4.18 and the hot fins are made by Acier inox Z10 as in Table 4.19. Use the macroscopic displacement fields and strain fields obtained from structural analysis, we can

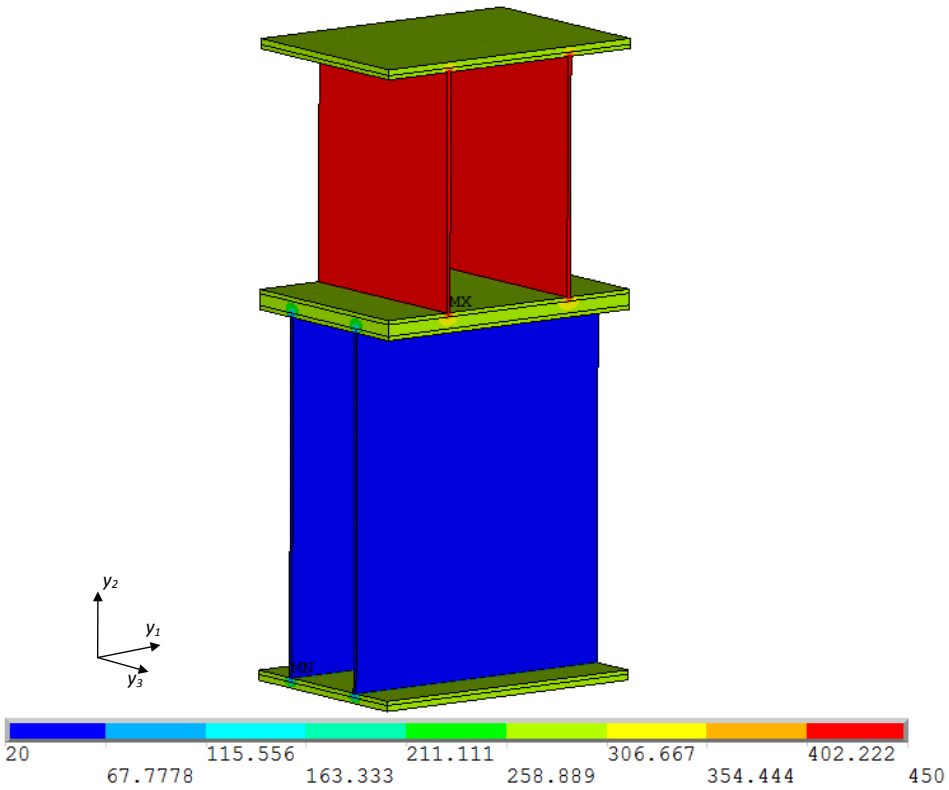

Fig. 4.12: Microstructure of heat exchanger fins under nonuniform temperature field

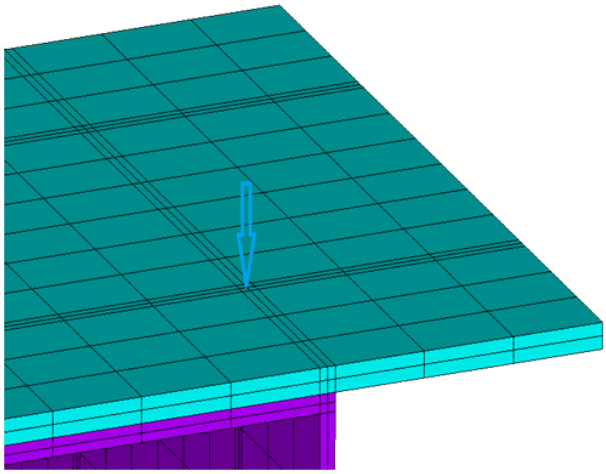

Fig. 4.13: Illustration of the location where results data obtained

Table 4.20: Temperature dependent material properties for Nickel 201

T(°C)	E_3(GPa)	ν_3	$\alpha_3(\mu/°C)$	ρ_3(kg/m^3)	K_3(W/m-°C)
20	204.994	0.290	12.50	8890	79.3
100	200.000	0.280	13.30	8890	79.3
150	197.500	0.285	13.60	8890	79.3
200	195.000	0.290	13.90	8890	79.3
250	192.500	0.290	14.05	8890	79.3
300	190.000	0.290	14.20	8890	79.3
350	186.500	0.285	14.50	8890	79.3
400	183.000	0.280	14.80	8890	79.3
450	180.030	0.280	15.05	8890	79.3
500	177.059	0.280	15.30	8890	79.3

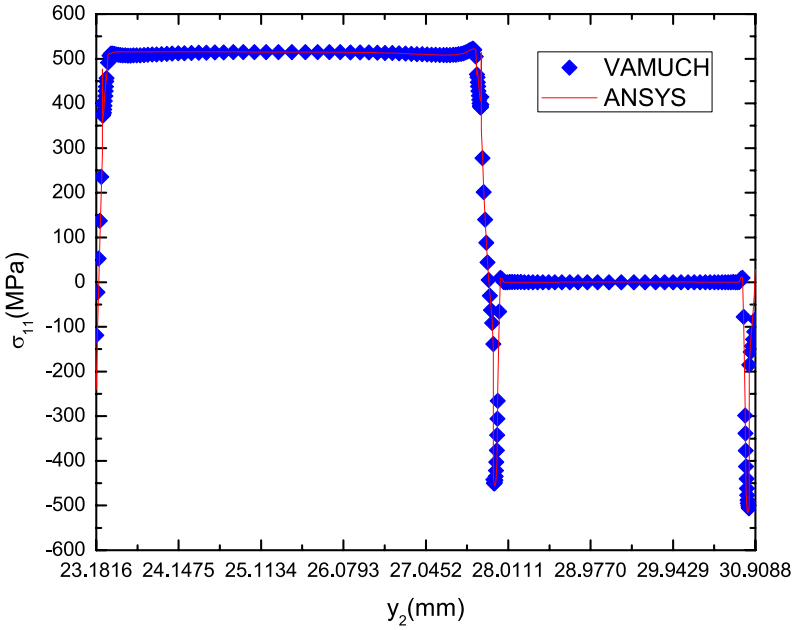

Fig. 4.14: Comparison of thermal stress σ_{11} along y_2 at $y_1 = 0.905$mm and $y_3 = 0.645$mm

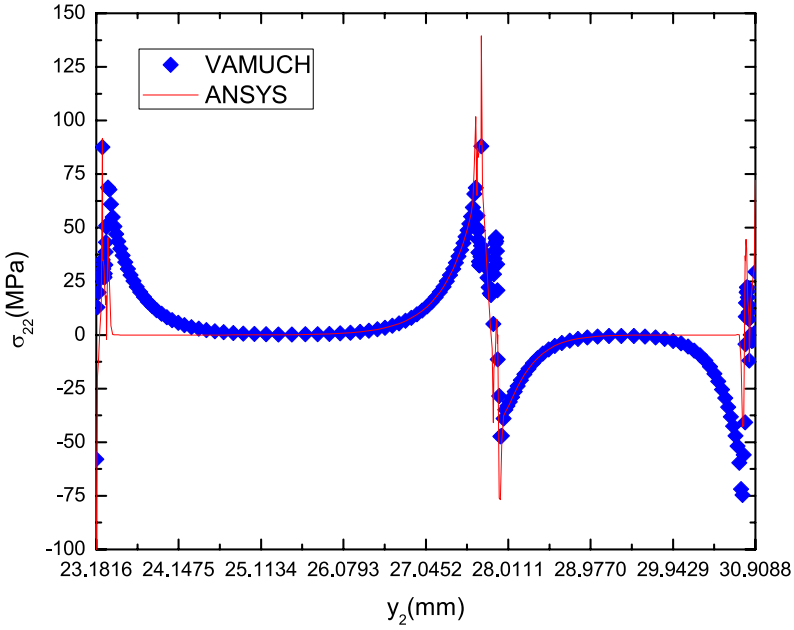

Fig. 4.15: Comparison of thermal stress σ_{22} along y_2 at $y_1 = 0.905$mm and $y_3 = 0.645$mm

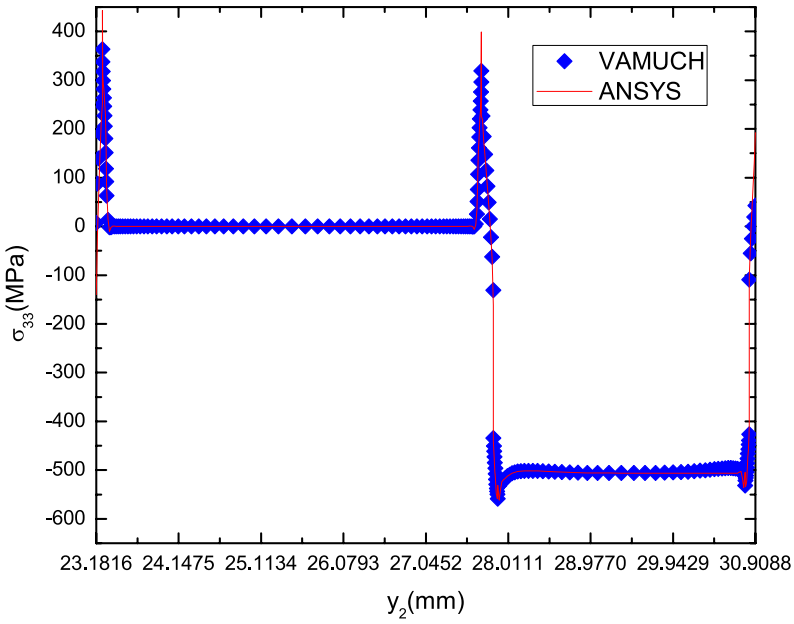

Fig. 4.16: Comparison of thermal stress σ_{33} along y_2 at $y_1 = 0.905$mm and $y_3 = 0.645$mm

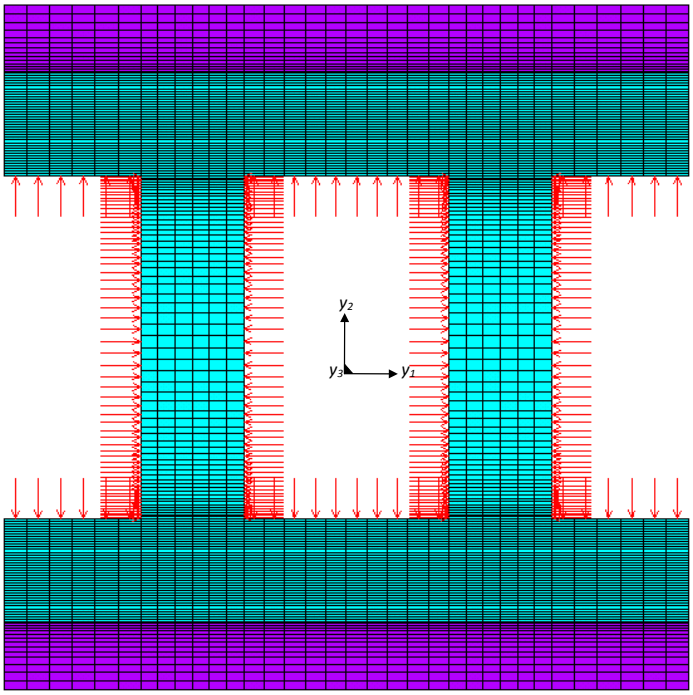

Fig. 4.17: Microstructure of a two-phase composite with pressure loads

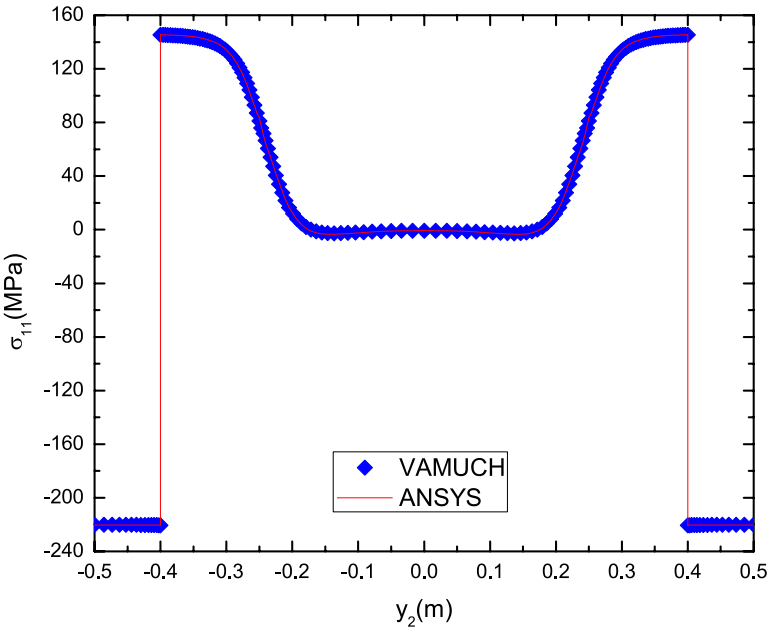

Fig. 4.18: Comparison of local stress σ_{11} along y_2 at $y_1 = 0.225$m and $y_3 = 0.25$m

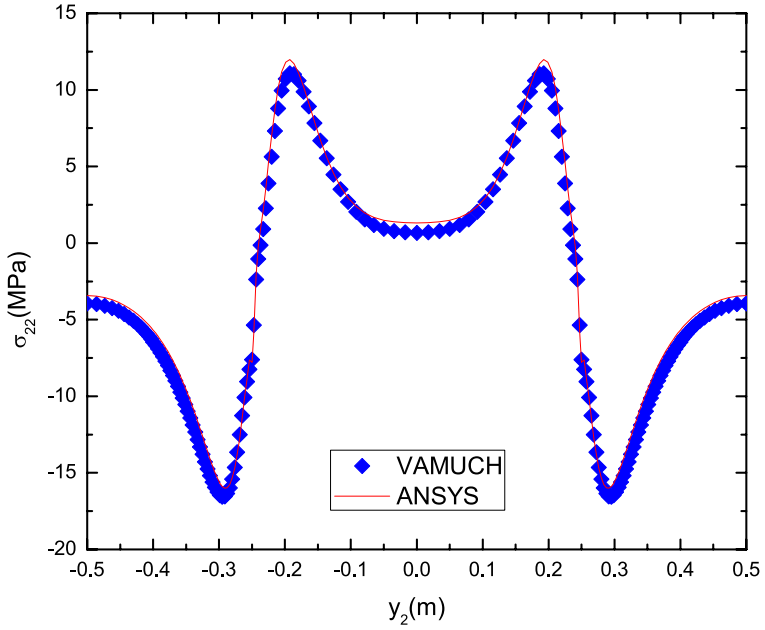

Fig. 4.19: Comparison of local stress σ_{22} along y_2 at $y_1 = 0.225$m and $y_3 = 0.25$m

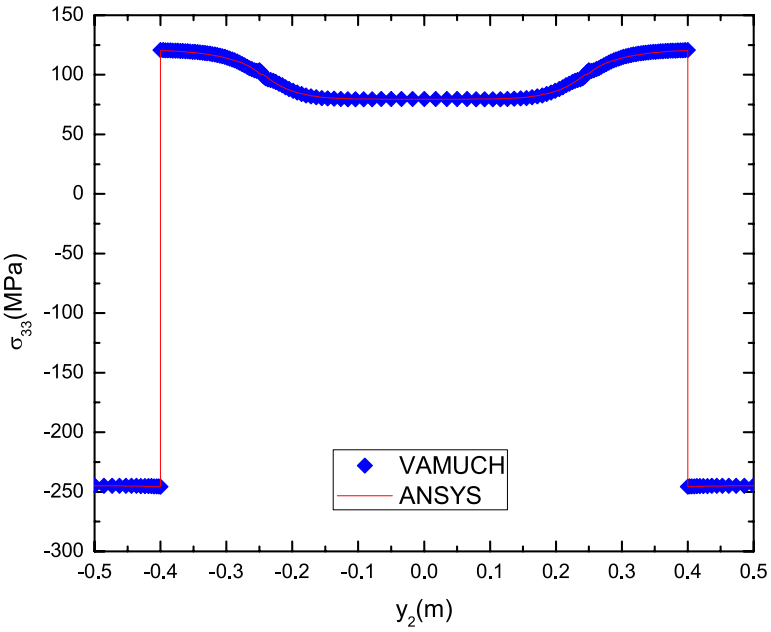

Fig. 4.20: Comparison of local stress σ_{33} along y_2 at $y_1 = 0.225$m and $y_3 = 0.25$m

recover the local stresses of this hot cell. Again, in order to obtain better results, we make mesh of the hot cell model as dense as that our current computational power can handle. The stress results are plotted along y_2 direction at the location where $y_1 = 13.575$mm and $y_3 = 1.935$mm in Figs. 4.23, 4.24, and 4.25. It is easy to notice that comparing with stress distribution in 22 direction for double cell because of the denser mesh applied here, the σ_{22} result is closer to each other between ANSYS and VAMUCH, in the mean time, good matches are observed for σ_{11} and σ_{33} results.

4.4.6 Aerospace Heat exchanger Fins under Working Conditions

Last, we conduct a micromechanical analysis of a unit cell of aerospace heat exchanger fins under its working condition which experiences consecutive hot air of 450°C going through top layer and consecutive cold air of 20°C blowing in from bottom layer. The heat exchanger fins are formed by repeating this unit cell structure many times in all three directions. The temperature field is similar like we showed in Fig. 4.12 except this unit cell has doubled the thickness of middle frames in order to reduce the unnecessary error induced by mesh. The resulting pressure loads caused by those two consecutive flows of air are showed in Fig. 4.26. The hot cell layer on top experiences a pressure load of 0.6MPa on all inner surfaces and the cold cell layer on bottom experiences a pressure load of 0.2MPa on all inner surfaces. Again, we assume the materials are stress and strain free at 20°C. Use the macroscopic displacement fields and strain fields obtained from structural analysis, we can recover the local stresses of this hot cell. The stress results are plotted along y_2 direction at the location where $y_1 = -0.88$mm and $y_3 = 0.6196$mm in Figs. 4.27, 4.28, and 4.29. Moreover, this position is same as the position we used for only nonuniformly distributed temperature applied case in Fig. 4.13 though the numbers of y_1 and y_3 are changed. We observed perfect matches for σ_{11} and σ_{33} results while for σ_{22} we obtained similar results as shown in the case only with nonuniformly distributed temperature fields. Again, the ANSYS result of σ_{22} at the regions close to top and bottom parting sheets does not perform a very smooth transition of stress as shown in Fig 4.28 while VAMUCH result still perform a smooth transition. For the sense of this microstructure and corresponding temperature

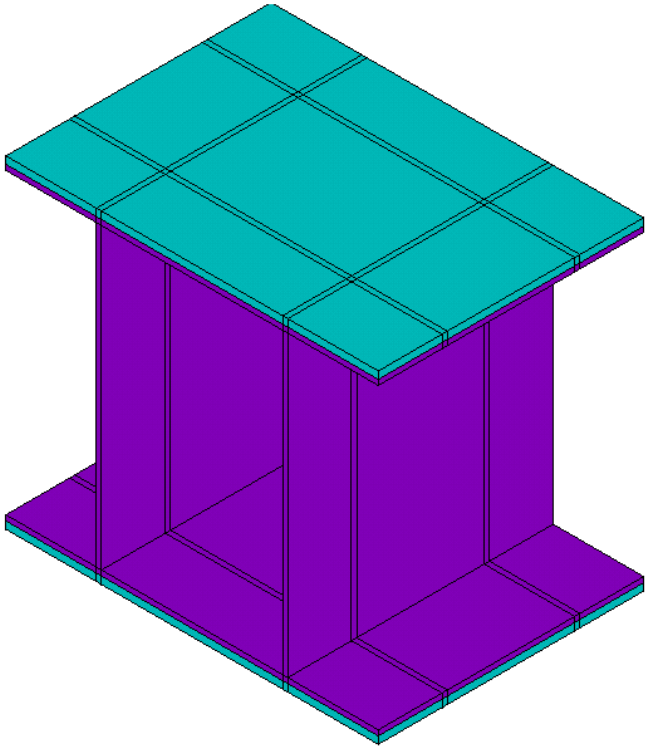

Fig. 4.21: Sketch of a hot cell

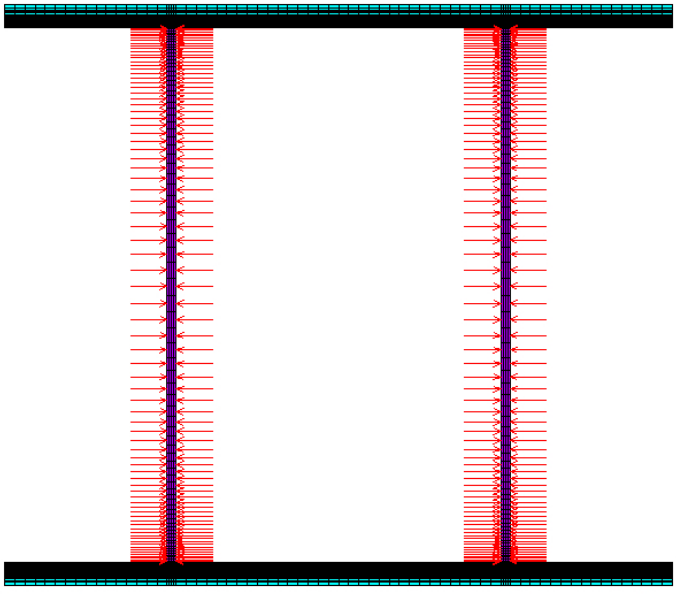

Fig. 4.22: A hot cell with pressure loads

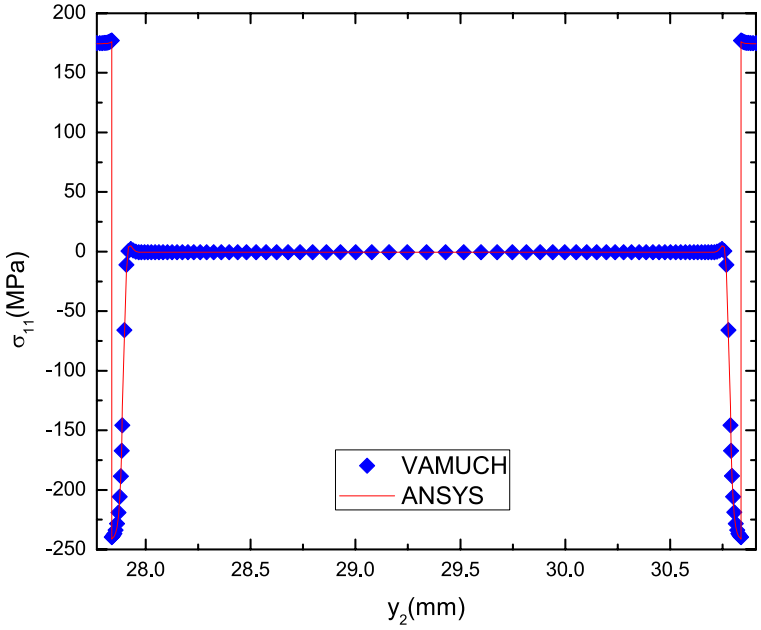

Fig. 4.23: Comparison of local stress σ_{11} along y_2 at $y_1 = 13.575$mm and $y_3 = 1.935$mm

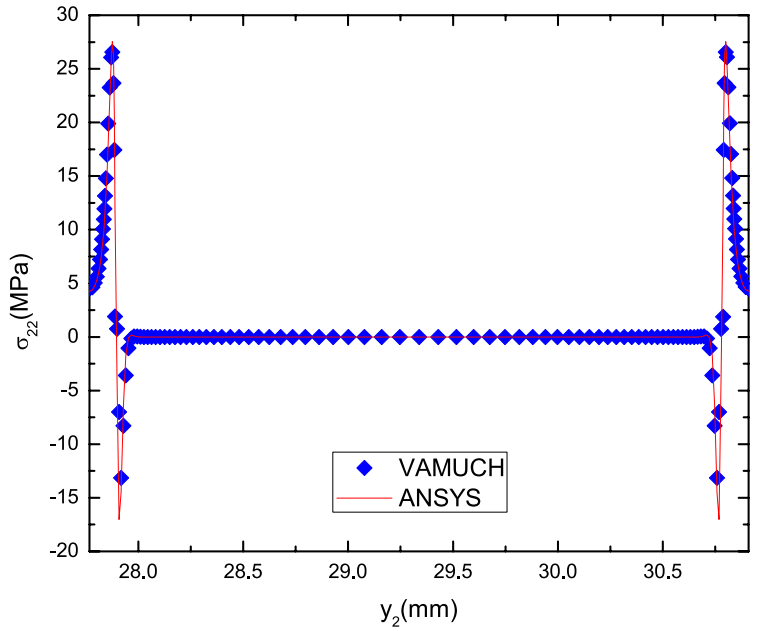

Fig. 4.24: Comparison of local stress σ_{22} along y_2 at $y_1 = 13.575$mm and $y_3 = 1.935$mm

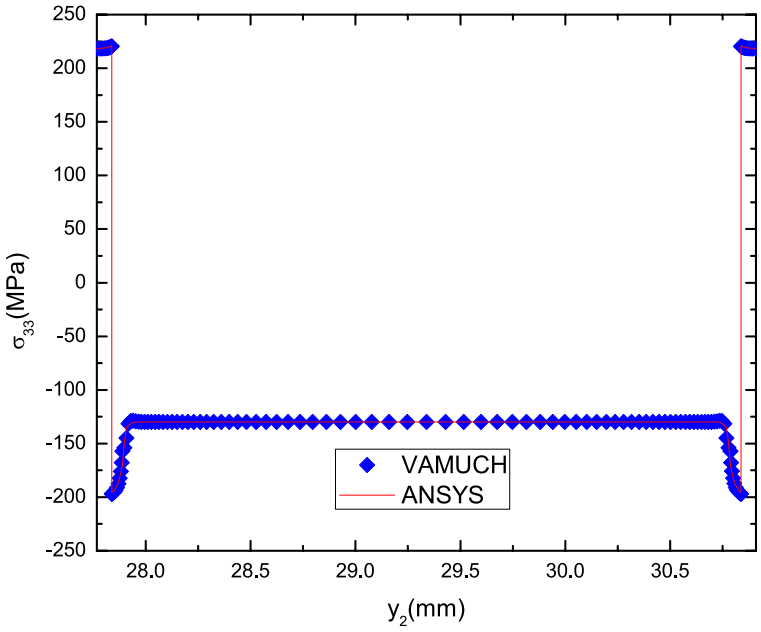

Fig. 4.25: Comparison of local stress σ_{33} along y_2 at $y_1 = 13.575$mm and $y_3 = 1.935$mm

fields, we believe VAMUCH result is more trustful in this situation.

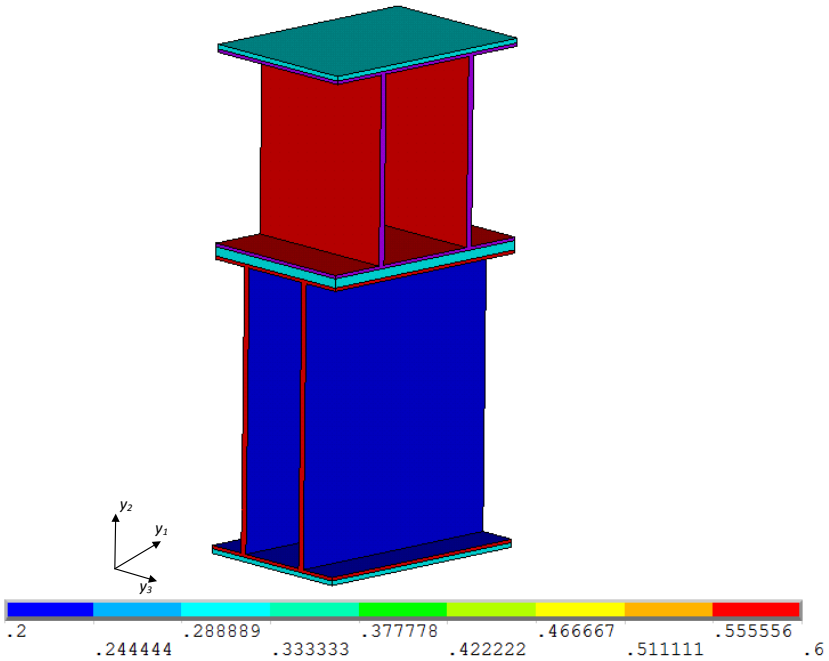

Fig. 4.26: Microstructure of heat exchanger fins under pressure loads

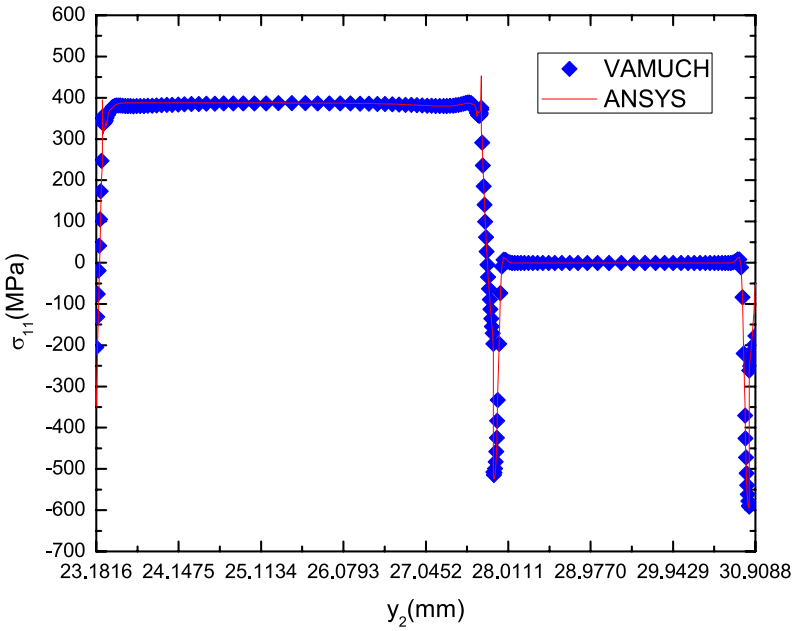

Fig. 4.27: Comparison of local stress σ_{11} along y_2 at $y_1 = -0.88$mm and $y_3 = 0.6196$mm

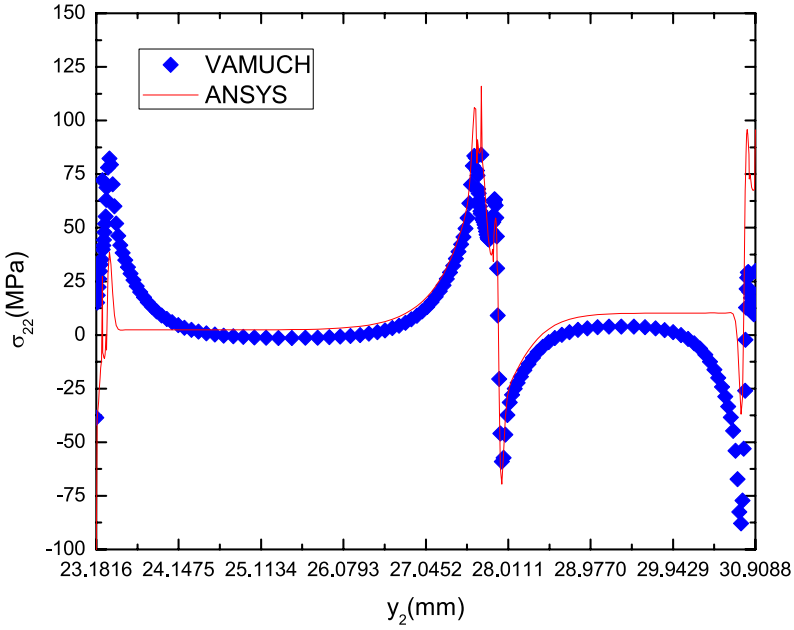

Fig. 4.28: Comparison of local stress σ_{22} along y_2 at $y_1 = -0.88$mm and $y_3 = 0.6196$mm

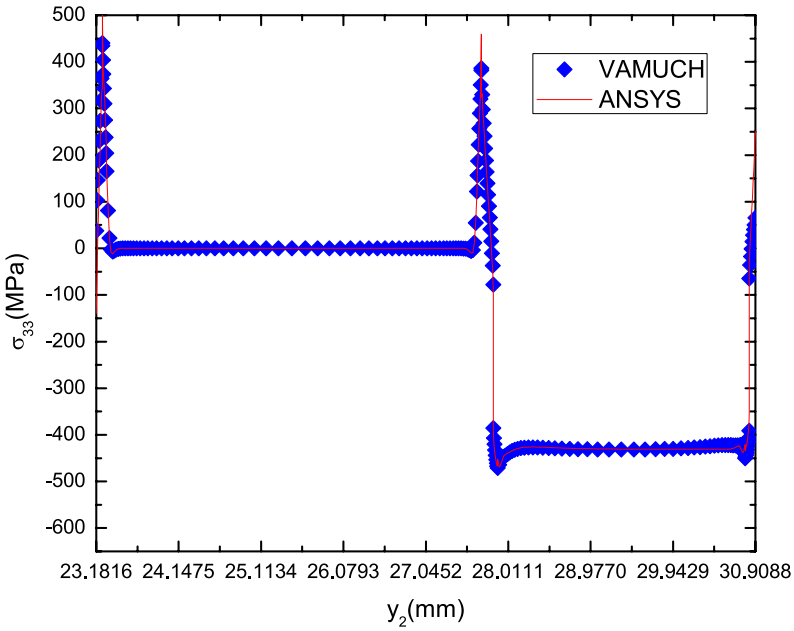

Fig. 4.29: Comparison of local stress σ_{33} along y_2 at $y_1 = -0.88$mm and $y_3 = 0.6196$mm

Chapter 5

Conclusions and Future Works

The current research focuses on developing new thermomechanical micromechanics models handling temperature dependent material properties, large temperature variation, nonuniformly distributed temperature field, and both internal and external loads may exist in the microstructure. Most of the limitations of current linear thermoelastic theory have been overcome by applying the newly developed thermomechanical micromechanics models. They are extended based on previous linear thermoelastic micromechanics model implemented inside VAMUCH. In this chapter, we will give a review of the accomplishments of this dissertation and make suggestions to the related future research.

5.1 Conclusions

The VAMUCH based thermomechanical micromechanics analysis has been extended in the current study by abandoning the traditional linear theory of thermoelasticity. Kavalenko's theory of small-strain thermoelasticity has been adopted and developed in order to remove the restrictions of small temperature variation within the microstructure. This requires VAMUCH to eliminate current restriction on temperature change and also be able to handle temperature dependent material properties as most of the materials in nature will perform differently with temperatures that are greatly changed.

For this problem, in the upgraded version of VAMUCH, like most of the commercial FEA software such as ANSYS, users are able to choose applying either secant coefficients of thermal expansions or instantaneous coefficients of thermal expansions as their material property inputs to adopt the thermomechanical micromechanics analysis with large or small temperature variations. Also this new thermomechanical micromechanics model enables VAMUCH to give effective material property results at each temperature by running the

micromechanical analysis only once.

VAMUCH provides a one way thermomechanical coupling for thermomechanical micromechanics analysis. This requires the temperature field to be known before every analysis and only uniform temperature field is taken in the previous model. In reality, most of the engineering structures will not experience uniform temperature fields during their working conditions. To this end, we removed this limitation by enabling VAMUCH to handle nonuniformly distributed temperature field. This ability greatly enhances the capability of VAMUCH dealing with realistic thermomechanical problems and dramatically save efforts of users to run this kind of analysis. To be more specific, if one is interested in using VAMUCH to calculate effective material properties or recover local stress and strain fields, he/she only needs to apply corresponding thermal loads and boundaries into commercial FEA software to carry on a heat conduction or convection analysis. After that, VAMUCH will read into the resulting temperature field no matter it is uniform or nonuniform and give the corresponding results for such problem. However, before this new approach, it is almost impossible to connect VAMUCH with heat conduction or convection analysis unless the temperature field is simply uniform.

Also the traditional micromechanics analysis has an assumption of load-free within the microstructure which may not always be true for real practical engineering problems. For examples, gas turbine blades and heat exchanger fins are usually experiencing air pressures during their operating conditions. By applying the principle of virtual work, we developed a new micromechanics model which takes into account of both internal and external loads. This newly developed micromechanics model enables VAMUCH to handle microstructures with loads which makes VAMUCH's capability of handling realistic engineering problems much more stronger than before.

At last, a HyperWorks-SwiftComp Micromechanics (aka VAMUCH) use interface is developed to connect HyperWorks preprocessor and postprocessor with SwiftComp Micromechanics. The user interface is capable of creating the standard composite models, applying the periodic boundary conditions, adding or editing constituents material proper-

ties, generating SwiftComp Micromechanics inputs, and running micromechanics analysis and providing results. The detailed introduction and demonstration of this interface are listed in Appendix A.

5.2 Recommendations for Future Work

As we mentioned above, the one-way thermomechanical coupling is used in VAMUCH for thermomechanical micromechanics analysis which requires users to run a heat conduction or convection analysis before using VAMUCH. This may not be very convenient and it is easy to induce mistakes when transferring resulting temperature fields into VAMUCH for micromechanics analysis. In the future study, fully coupled thermomechanical problem are recommended in order to avoid these extra efforts of obtaining temperature fields. Moreover, for studying large temperature variation, even though the coefficients of thermal expansions are usually small comparing with elastic constants, there is still possibility of the total strain induced by both mechanical and thermal loads exceed the limit of small strain. In this case, we can not analyze this kind of problems by applying small strain theory. I recommend future researchers working on large deformation problem, plasticity problem, and damage problem step by step.

Moreover, as aforementioned examples in Chapter 4, mesh effect is very critical for stress results of thin-walled structures such as the parting sheet of heat exchanger fins and gas turbine blades. VAMUCH has the capability of applying shell elements after 3.0 version, so I recommend developing a new thermomechanical micromechanics model using shell elements under nonuniformly distributed temperature fields for cases like the unit cell of heat exchanger fins to see if the results will be better than the current ones. Also if we can achieve that, large amount of computational efforts can be saved. For the cases with internal and external loads, we only tested periodic loads without applying on the periodic edges. We are not sure if the non-periodic loads or loads applying on the periodic edges will induce any problems. But it is definitely worth of investigation and we recommend researchers pay attention to these issues.

At last, the HyperWorks-SwiftComp Micromechanics User Interface is freshly developed and in version 1.0 now. If needed, more functionalities can be added such as create random unit cell by volume fractions, read and modify existing mesh of unit cell, add dummy boundary for unit cells with irregular shapes, and so on. Developers can refer to the source code and introduction listed in Appendix A.

References

[1] Voigt, W., "The Relation between the Two Elastic Moduli of Isotropic Materials," *Annals of Physics,(Leipzig)*, Vol. 33, 1889, pp. 573.

[2] Reuss, A., "Berechnung der Fließgrenze von Mischkristallen auf Grund der Plastizitätsbedingung für Einkristalle." *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, Vol. 9, No. 1, 1929, pp. 49–58.

[3] Hill, R., "The Elastic Behaviour of a Crystalline Aggregate," *Proceedings of the Physical Society. Section A*, Vol. 65, No. 5, 1952, pp. 349.

[4] Hashin, Z. and Shtrikman, S., "Note on a Variational Approach to the Theory of Composite Elastic Materials," *Journal of the Franklin Institute*, Vol. 271, No. 4, 1961, pp. 336–341.

[5] Hashin, Z. and Shtrikman, S., "A Variational Approach to the Theory of the Elastic Behaviour of Multiphase Materials," *Journal of the Mechanics and Physics of Solids*, Vol. 11, No. 2, 1963, pp. 127–140.

[6] Willis, J., "Bounds and Self-Consistent Estimates for the Overall Properties of Anisotropic Composites," *Journal of the Mechanics and Physics of Solids*, Vol. 25, No. 3, 1977, pp. 185–202.

[7] Beran, M. and Molyneux, J., "Use of Classical Variational Principles to Determine Bounds for the Effective Bulk Modulus in Heterogeneous Media," *Quarterly of Applied Mathematics*, Vol. 24, No. 2, 1966.

[8] McCOY, J. J., "On the Displacement Field in an Elastic Medium with Random Variations of Material Properties," *Recent Advances in Engineering Sciences*, Vol. 5, 1970, pp. 235–254.

[9] Silnutzer, N. R., *Effective Constants of Statistically Homogeneous Materials*, Ph.D. thesis, Graduate School of Arts and Sciences, University of Pennsylvania, 1972.

[10] Milton, G., "Bounds on the Complex Permittivity of a Two-Component Composite Material," *Journal of Applied Physics*, Vol. 52, No. 8, 1981, pp. 5286–5293.

[11] Milton, G., "Bounds on the Elastic and Transport Properties of Two-Component Composites," *Journal of the Mechanics and Physics of Solids*, Vol. 30, No. 3, 1982, pp. 177–191.

[12] Milton, G., "Bounds on the Transport and Optical Properties of a Two-Component Composite Material," *Journal of Applied Physics*, Vol. 52, No. 8, 1981, pp. 5294–5304.

[13] Milton, G., "Bounds on the Electromagnetic, Elastic, and Other Properties of Two-Component Composites," *Physical Review Letters*, Vol. 46, No. 8, 1981, pp. 542.

[14] Berryman, J. G., "Variational Bounds on Elastic Constants for the Penetrable Sphere Model," *Journal of Physics D: Applied Physics*, Vol. 18, No. 4, 1985, pp. 585.

[15] Berryman, J. G., "Interpolating and Integrating Three-Point Correlation Functions on a Lattice," *Journal of Computational Physics*, Vol. 75, No. 1, 1988, pp. 86–102.

[16] Berryman, J. G., "Estimating Effective Moduli of Composites Using Quantitative Image Analysis," *Random Media and Composites, Kohn, RV, and Milton, GW, eds., Soc. Ind. Appl. Math., Philadelphia*, 1989, pp. 3–12.

[17] Torquato, S., "Random Heterogeneous Media: Microstructure and Improved Bounds on Effective Properties," *Applied Mechanics Reviews*, Vol. 44, 1991, pp. 37.

[18] Miller, M. N., "Bounds for Effective Electrical, Thermal, and Magnetic Properties of Heterogeneous Materials," *Journal of Mathematical Physics*, Vol. 10, 1969, pp. 1988.

[19] Milton, G. and Phan-Thien, N., "New Bounds on Effective Elastic Moduli of Two-Component Materials," *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, Vol. 380, No. 1779, 1982, pp. 305–331.

[20] Torquato, S. and Stell, G., "Microstructure of Two-Phase Random Media. I. The N-point Probability Functions," *The Journal of Chemical Physics*, Vol. 77, 1982, pp. 2071.

[21] Torquato, S. and Stell, G., "Microstructure of Two-Phase Random Media. III. The N-point Matrix Probability Functions for Fully Penetrable Spheres," *The Journal of Chemical Physics*, Vol. 79, 1983, pp. 1505.

[22] Torquato, S., *Random Heterogeneous Materials: Microstructure and Macroscopic Properties*, Vol. 16, Springer-Verlag, New York, 2002.

[23] Duhamel, J. M. C., "Mémoire sur le Calcul des Actions Moléculaires Développées par les Changements de Température dans les Corps Solides," *Memoirs par Divers Savans (Acad. Sci. Paris)*, Vol. 5, 1838, pp. 440–498.

[24] Mase, G. T. and Mase, G. E., *Continuum Mechanics for Engineers*, CRC Press, Boca Raton, Florida, 2010.

[25] Chung, T., *Applied Continuum Mechanics*, Cambridge University Press, New York, 1996.

[26] Thomson, W., "On Thermo-Elastic and Thermo-Magnetic Properties of Matter," *Quarterly Journal of Mathematics*, Vol. 1, 1857, pp. 57–77.

[27] Rao, Y., *Chemical Engineering Thermodynamics*, Universities Press, Hyderabad, 1997.

[28] De Groot, S. R. and De Groot, S. R., *Thermodynamics of Irreversible Processes*, Amsterdam, North-Holland, 1951.

[29] Callen, H. B. and Greene, R. F., "On a Theorem of Irreversible Thermodynamics," *Physical Review*, Vol. 86, 1952, pp. 702–710.

[30] Biot, M. A., "Thermoelasticity and Irreversible Thermodynamics," *Journal of Applied Physics*, Vol. 27, No. 3, 1956, pp. 240–253.

[31] Prigogine, I., *Introduction to Thermodynamics of Irreversible Processes*, Interscience Publishers, New York, 3rd ed., 1967.

[32] Jou, D., Casas-Vázquez, J., and Lebon, G., *Extended Irreversible Thermodynamics*, Springer-Verlag Berlin Heidelberg, Berlin, 1996.

[33] Speziale, C., "On the Coupled Heat Equation of Linear Thermoelasticity," *Acta Mechanica*, Vol. 150, No. 1-2, 2001, pp. 121–126.

[34] Kovalenko, A., "Fundamentals of Thermoelasticity," *Strength of Materials*, Vol. 3, No. 9, 1971, pp. 1134–1135.

[35] Nadeau, J. and Ferrari, M., "Effective Thermal Expansion of Heterogeneous Materials with Application to Low Temperature Environments," *Mechanics of Materials*, Vol. 36, No. 3, 2004, pp. 201–214.

[36] Chamis, C. and Sendeckyj, G., "Critique on Theories Predicting Thermoelastic Properties of Fibrous Composites," *Journal of Composite Materials*, Vol. 2, No. 3, 1968, pp. 332–358.

[37] Peterson, G. and Fletcher, L., "A Review of Thermal Conductivity in Composite Materials," *Rep./AIAA*, Vol. 87, 1987, pp. 1586.

[38] Dvorak, G. J., "Plasticity theories for fibrous composites," *Metal Matrix Composites: Mechanisms and Properties*, Academic Press, Boston, 1991, pp. 1–77.

[39] Noor, A. K. and Shah, R. S., "Effective Thermoelastic and Thermal Properties of Unidirectional Fiber-Reinforced Composites and Their Sensitivity Coefficients," *Composite Structures*, Vol. 26, No. 1, 1993, pp. 7–23.

[40] Agarwal, B. D., Broutman, L. J., and Chandrashekhara, K., *Analysis and Performance of Fiber Composites*, Wiley. com, 2006.

[41] Greszczuk, L., "Thermoelastic Properties of Filamentary Composites," *Proceedings of the AIAA/ASME 6th Structures, Structural Dynamics, and Materials Conference, Palm Springs, California*, 1965.

[42] Abolin'sh, D., "Compliance Tensor for an Elastic Material Reinforced in One Direction," *Polymer Mechanics*, Vol. 1, No. 4, 1965, pp. 28–32.

[43] Chamis, C. C., "Simplified Composite Micromechanics Equations of Hygral, Thermal, and Mechanical Properties," *SAMPE Quarterly*, Vol. 15, 1984, pp. 14–23.

[44] Caruso, J. J. and Chamis, C. C., "Assessment of Simplified Composite Micromechanics Using Three-Dimensional Finite-Element Analysis," *Journal of Composites Technology and Research*, Vol. 8, 1986, pp. 77–83.

[45] Hopkins, D. A. and Chamis, C. C., "A Unique Set of Micromechanics Equations for High-Temperature Metal Matrix Composites," *ASTM STP*, Vol. 964, 1988, pp. 159–176.

[46] Jones, R. M., *Mechanics of Composite Materials*, Hemispheres Publishing, New York, 1975.

[47] Christensen, R. M., *Mechanics of Composite Materials*, DoverPublications. com, 2012.

[48] Mital, S., Murthy, P., and Chamis, C., "Micromechanics for Ceramic Matrix Composites via Fiber Substructuring," *Journal of Composite Materials*, Vol. 29, No. 5, 1995, pp. 614–633.

[49] Eshelby, J. D., "The Determination of the Elastic Field of an Ellipsoidal Inclusion, and Related problems," *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, Vol. 241, No. 1226, 1957, pp. 376–396.

[50] Hershey, A., "The Elasticity of an Isotropic Aggregate of Anisotropic Cubic Crystals," *Journal of Applied Mechanics*, Vol. 21, No. 3, 1954, pp. 226–240.

[51] Kröner, E., "Berechnung der elastischen Konstanten des Vielkristalls aus den Konstanten des Einkristalls," *Zeitschrift für Physik*, Vol. 151, No. 4, 1958, pp. 504–518.

[52] Hill, R., "A Self-Consistent Mechanics of Composite Materials," *Journal of the Mechanics and Physics of Solids*, Vol. 13, No. 4, 1965, pp. 213–222.

[53] Hill, R., "Theory of Mechanical Properties of Fibre-Strengthened MaterialsIII. Self-Consistent Model," *Journal of the Mechanics and Physics of Solids*, Vol. 13, No. 4, 1965, pp. 189–198.

[54] Budiansky, B., "On the Elastic Moduli of Some Heterogeneous Materials," *Journal of the Mechanics and Physics of Solids*, Vol. 13, No. 4, 1965, pp. 223–227.

[55] Budiansky, B., "Thermal and Thermoelastic Properties of Isotropic Composites," *Journal of Composite Materials*, Vol. 4, No. 3, 1970, pp. 286–295.

[56] Laws, N., "On the Thermostatics of Composite Materials," *Journal of the Mechanics and Physics of Solids*, Vol. 21, No. 1, 1973, pp. 9–17.

[57] Banerjee, B. and Adams, D. O., "Micromechanics-Based Prediction of Thermoelastic Properties of High Energy Materials," *Constitutive Modeling of Geomaterials*, 2003, pp. 158–164.

[58] Mori, T. and Tanaka, K., "Average Stress in Matrix and Average Elastic Energy of Materials with Misfitting Inclusions," *Acta Metallurgica*, Vol. 21, No. 5, 1973, pp. 571–574.

[59] Benveniste, Y., "A New Approach to the Application of Mori-Tanaka's Theory in Composite Materials," *Mechanics of Materials*, Vol. 6, No. 2, 1987, pp. 147–157.

[60] Weng, G., "The Theoretical Connection between Mori-Tanaka's Theory and the Hashin-Shtrikman-Walpole Bounds," *International Journal of Engineering Science*, Vol. 28, No. 11, 1990, pp. 1111–1120.

[61] Berryman, J. G. and Berge, P. A., "Critique of Two Explicit Schemes for Estimating Elastic Properties of Multiphase Composites," *Mechanics of Materials*, Vol. 22, No. 2, 1996, pp. 149–164.

[62] Kuster, G. T. and Toksöz, M. N., "Velocity and Attenuation of Seismic Waves in Two-Phase Media: Part I. Theoretical Formulations," *Geophysics*, Vol. 39, No. 5, 1974, pp. 587–606.

[63] Norris, A., "An Examination of the Mori-Tanaka Effective MediumApproximation for Multiphase Composites," *Journal of Applied Mechanics*, Vol. 56, 1989, pp. S3.

[64] Benveniste, Y., Chen, T., and Dvorak, G., "The Effective Thermal Conductivity of Composites Reinforced by Coated Cylindrically Orthotropic Fibers," *Journal of Applied Physics*, Vol. 67, No. 6, 1990, pp. 2878–2884.

[65] Böhm, H. J. and Nogales, S., "Mori–Tanaka Models for the Thermal Conductivity of Composites with Interfacial Resistance and Particle Size Distributions," *Composites Science and Technology*, Vol. 68, No. 5, 2008, pp. 1181–1187.

[66] Torquato, S. and Rintoul, M., "Effect of the Interface on the Properties of Composite Media," *Physical Review Letters*, Vol. 75, No. 22, 1995, pp. 4067.

[67] Aboudi, J., "Micromechanical Analysis of Composites by the Method of Cells," *Applied Mechanics Reviews*, Vol. 42, 1989, pp. 193.

[68] Aboudi, J., "A Continuum Theory for Fiber-Reinforced Elastic-Viscoplastic Composites," *International Journal of Engineering Science*, Vol. 20, No. 5, 1982, pp. 605–621.

[69] Aboudi, J., *Mechanics of Composite Materials-A Unified Micromechanical Approach*, Elsevier, Amsterdam and New York, 1991.

[70] Paley, M. and Aboudi, J., "Micromechanical Analysis of Composites by the Generalized Cells Model," *Mechanics of Materials*, Vol. 14, No. 2, 1992, pp. 127–139.

[71] Aboudi, J. and Pindera, M.-J., "High-Fidelity Micromechanical Modeling of Continuously Reinforced Elastic Multiphase Materials Undergoing Finite Deformations," *Mathematics and Mechanics of Solids*, Vol. 9, No. 6, 2004, pp. 599–628.

[72] Bednarcyk, B. A., Arnold, S. M., Aboudi, J., and Pindera, M.-J., "Local Field Effects in Titanium Matrix Composites Subject to Fiber-Matrix Debonding," *International Journal of Plasticity*, Vol. 20, No. 8, 2004, pp. 1707–1737.

[73] Carpenter, D. K., Santiago, G., and Hunt, A. H., "Aggregation of Polyoxyethylene in Dilute Solutions," *Journal of Polymer Science: Polymer Symposia*, Vol. 44, Wiley Online Library, 1974, pp. 75–92.

[74] Norris, A., "A Differential Scheme for the Effective Moduli of Composites," *Mechanics of Materials*, Vol. 4, No. 1, 1985, pp. 1–16.

[75] Markov, K. Z., "Elementary Micromechanics of Heterogeneous Media," *Heterogeneous Media*, Birkhuser Boston, Cambridge, Massachusetts, 2000, pp. 1–162.

[76] Williams, T. O., "A Two-Dimensional, Higher-Order, Elasticity-Based Micromechanics Model," *International Journal of Solids and Structures*, Vol. 42, No. 3, 2005, pp. 1009–1038.

[77] Williams, T. O., "A Three-Dimensional, Higher-Order, Elasticity-Based Micromechanics Model," *International Journal of Solids and Structures*, Vol. 42, No. 3-4, 2005, pp. 971–1007.

[78] Yu, W., Williams, T. O., Bednarcyk, B. A., Aboudi, J., and Tang, T., "A Critical Evaluation of the Predictive Capabilities of Various Advanced Micromechanics Models," *Proceedings of the 48th Structures, Structural Dynamics and Materials Conference, Waikiki, Hawaii*, 2007.

[79] Sun, C. and Vaidya, R., "Prediction of Composite Properties from a Representative Volume Element," *Composites Science and Technology*, Vol. 56, No. 2, 1996, pp. 171–179.

[80] Berdichevskii, V., "Variational-Asymptotic Method of Constructing a Theory of Shells: PMM," *Journal of Applied Mathematics and Mechanics*, Vol. 43, No. 4, 1979, pp. 711–736.

[81] Yu, W., "A Variational-Asymptotic Cell Method for Periodically Heterogeneous Materials," ASME, 2005.

[82] Yu, W. and Tang, T., "Variational Asymptotic Method for Unit Cell Homogenization of Periodically Heterogeneous Materials," *International Journal of Solids and Structures*, Vol. 44, No. 11, 2007, pp. 3738–3755.

[83] Yu, W. and Tang, T., "A Variational Asymptotic Micromechanics Model for Predicting Thermoelastic Properties of Heterogeneous Materials," *International Journal of Solids and Structures*, Vol. 44, No. 22, 2007, pp. 7510–7525.

[84] Lee, C.-Y. and Yu, W., "Variational Asymptotic Modeling of Composite Beams with Spanwise Heterogeneity," *Computers & Structures*, Vol. 89, No. 15, 2011, pp. 1503–1511.

[85] Boley, B. A. and Weiner, J. H., *Theory of Thermal Stresses*, Wiley, New York, 1960.

[86] Reddy, J. N., *An Introduction to Continuum Mechanics*, Cambridge University Press, 2008.

[87] Truesdell, C., *Continuum Mechanics: The Mechanical Foundations of Elasticity and Fluid Dynamics*, Gordon and Breach, New York, 1966.

[88] Moran, M. J., Shapiro, H. N., Boettner, D. D., and Bailey, M., *Fundamentals of Engineering Thermodynamics*, Wiley, Hoboken, New Jersey, 2011.

[89] Lord, H. W. and Shulman, Y., "A Generalized Dynamical Theory of Thermoelasticity," *Journal of the Mechanics and Physics of Solids*, Vol. 15, No. 5, 1967, pp. 299–309.

[90] Kovalenko, A., *Osnovy termouprugosti*, Naukova Dumka, Kiev, 1970.

[91] Lubarda, V., "On Thermodynamic Potentials in Linear Thermoelasticity," *International Journal of Solids and Structures*, Vol. 41, No. 26, 2004, pp. 7377–7398.

[92] Boussaa, D., "Effective Thermoelastic Properties of Composites with Temperature-Dependent Constituents," *Mechanics of Materials*, Vol. 43, No. 8, 2011, pp. 397–407.

[93] Sokolnikoff, I. S. and Specht, R. D., *Mathematical Theory of Elasticity*, McGraw-Hill, New York, 2nd ed., 1956.

[94] Berdichevsky, L. V., *Variational Principles of Continuum Mechanics*, Springer-Verlag Berlin Heidelberg, London and New York, 2009.

[95] ERG Aerospace Corporation, "Physical Characteristics of Duocel Silicon Carbide Foam*," http://www.ergaerospace.com/SiC-properties.htm, 2011.

[96] Miracle, D. B. and Donaldson, S. L., *ASM Handbook: Composites*, Vol. 21, ASM International, 2001.

Appendices

Appendix A

A User Interface of HyperWorks-SwiftComp Micromechanics (aka VAMUCH)

A.1 Introduction of HyperWorks-SwiftComp Micromechanics User Interface

HyperWorks-SwiftComp Micromechanics User Interface is created by the author during his Ph.D. study for facilitating HyperWorks users to convert HyperMesh geometrical model information and material properties into SwiftComp Micromechanics inputs as a preprocessor and to view results using HyperView as a postprocessor. It is acting as a bridge between HyperWorks and SwiftComp Micromechanics by sending the information back and forth. Users can click the corresponding buttons under the interface to perform SwiftComp Micromechanics micromechanical analysis and local stress and strain recovery analysis. Also classical 2D fiber reinforced composite model and 3D particle reinforced composite model are provided to reduce standard micromechanics analysis time.

HyperWorks-SwiftComp Micromechanics User Interface is programmed using TCL/TK script language and template commands under HyperWorks standards.

A.2 Functionalities of the Interface

After successfully installed the interface, starting the HyperWorks, you should be able to see a MicroModel button under Utility tab as shown in Fig. A.1. There are two major sections in this interface: Model Tools and Micro Analysis Tools. Model Tools provide tools to create and edit geometry and material properties of composites while Micro Analysis Tools provide tools to generate SwiftComp Micromechanics inputs for either micromechanical analysis or recovery analysis and to execute corresponding micromechanics analysis in order to generate results.

The Common Unit Cell button is used as one click to create standard 2D and 3D UCs with user favored size and material properties as shown in Fig. A.2. A sample 2D fiber reinforced standard unit cell with 0.5 fiber volume fraction can be created using this button as shown in Fig. A.3. The Custom Unit Cell button is used to apply the periodic boundary conditions onto the geometry as SwiftComp Micromechanics requires users to provide periodic boundary information when the model center is not at origin of the coordinate. Two ways of applying periodic boundary condition are provided. Automatic is used to add periodic boundary by dimensions which is basically for models with regular geometry shapes as shown in Fig. A.4 while Manual is used to manually add periodic boundary which is basically for models with complicated shapes or irregular shapes as shown in Fig. A.5. Material Properties button is used for users to view and edit material properties which is more straight forward and easier to operate than the material cards provided by HyperWorks as shown in Figs. A.6 and A.7. This button is directly borrowed from ANSYS-HyperWorks interface developed by other code developers. Analysis Type is used to choose the micro analysis type and generate the corresponding input files as shown in Figs. A.8 and A.9 while Solve button calls SwiftComp Micromechanics to solve the problems and shows the outputs for micromechanical analyses directly as shown in Fig. A.10. The recovered local displacement fields, strain fields, and stress fields can be viewed by importing the result files to HyperView as shown in Fig. A.11.

A.3 The Coding Structure of the Interface

The coding structure of HyperWorks-SwiftComp Micromechanics User Interface starts from a new *hm_micro.mac* created within HyperWorks which users need to apply this macro file under Preference menu following installation guide. The original functionalities of hm.mac (default) are also included in this macro file so users do not need to worry about losing the existing functionalities. A new *globalpage.mac* file is created for adding the micro analysis button into Utility tab. The main command file of the interface is included in *micro.mac* where *commonUC.tcl*, *customUC.tcl*, *material.tcl*, *microtype.tcl*, and *microsolve.tcl* are functioned there to connect with each button of the interface respec-

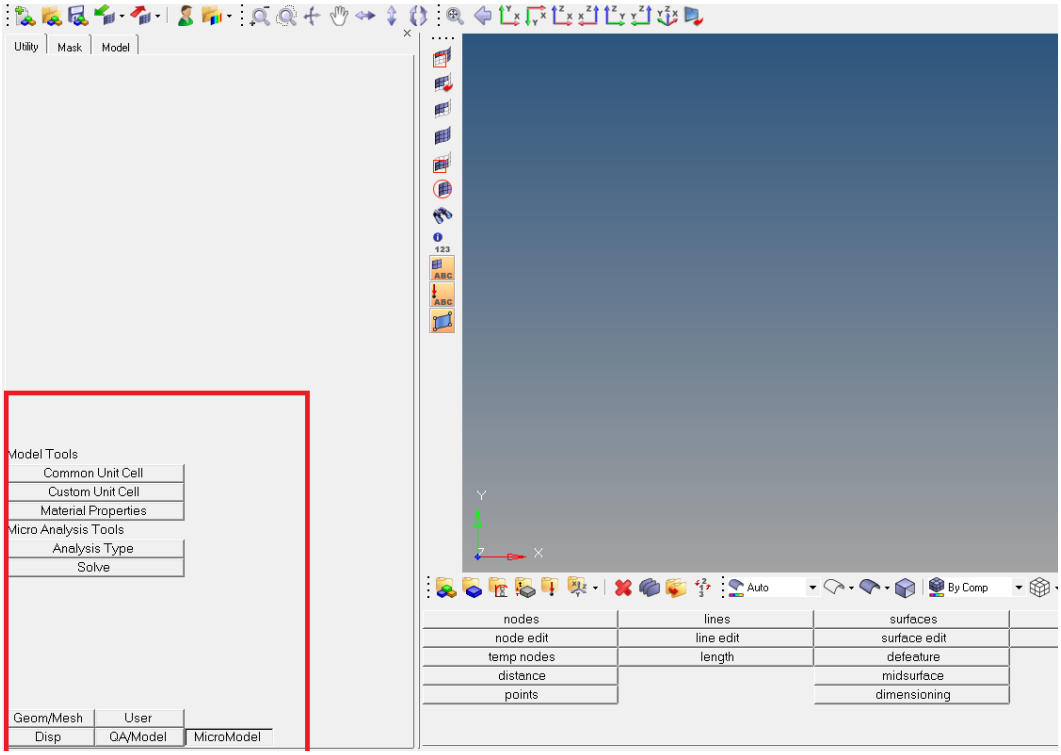

Fig. A.1: Sketch of HyperWorks-SwiftComp Micromechanics User Interface 1

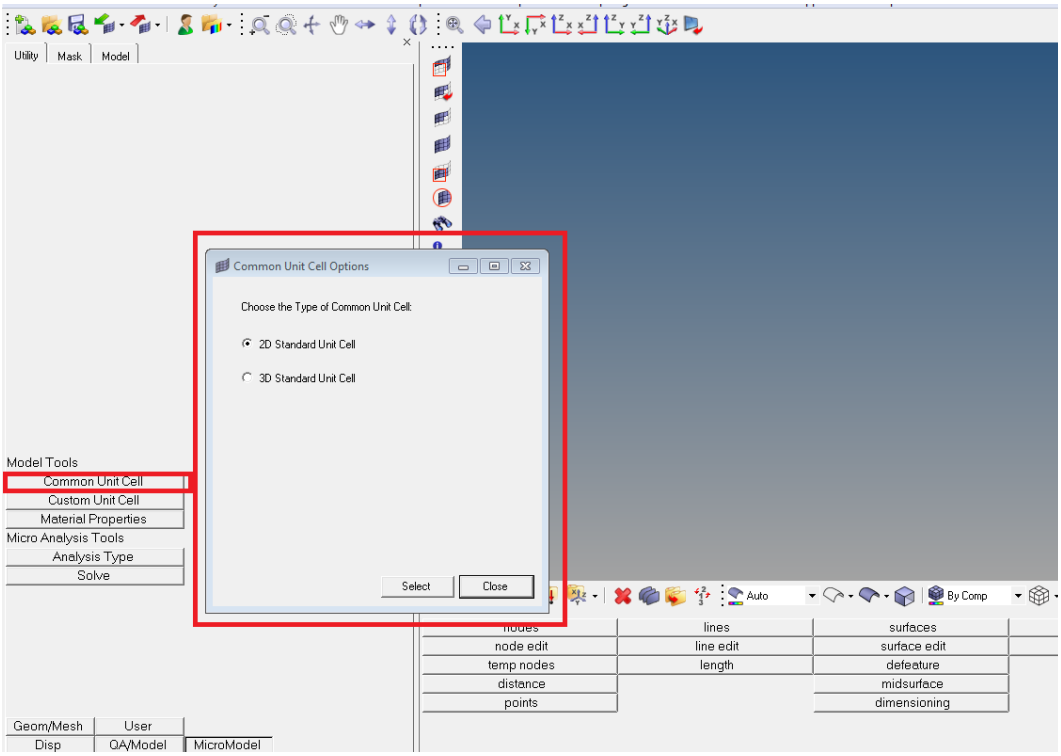

Fig. A.2: Sketch of HyperWorks-SwiftComp Micromechanics User Interface 2

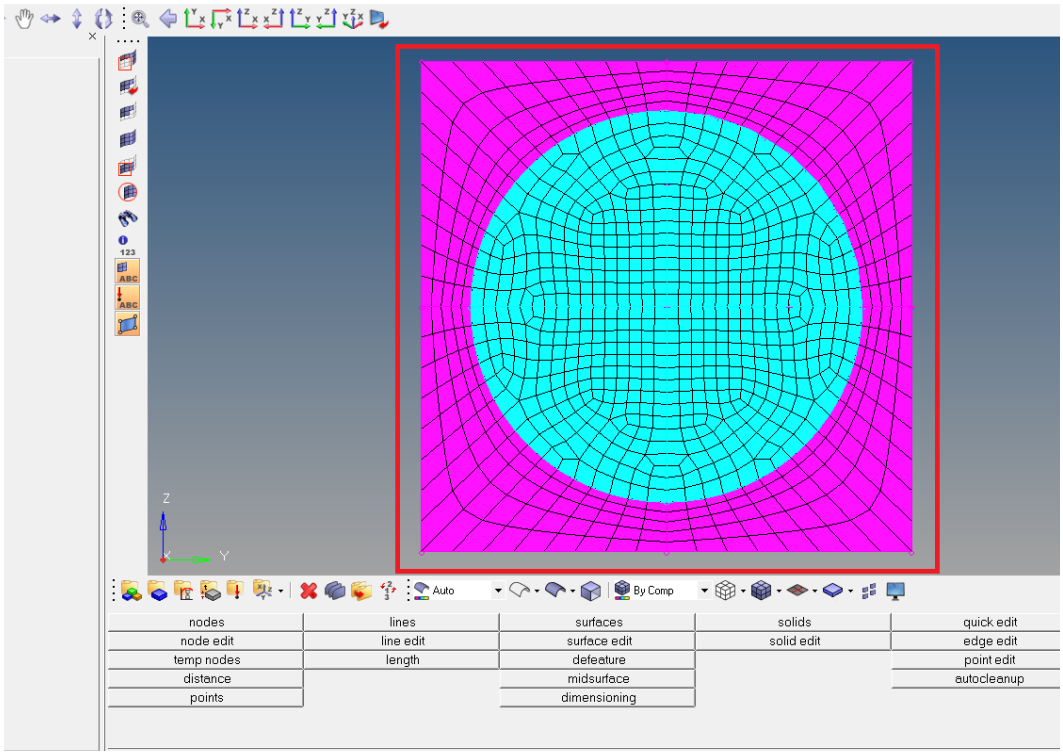

Fig. A.3: Sketch of HyperWorks-SwiftComp Micromechanics User Interface 3

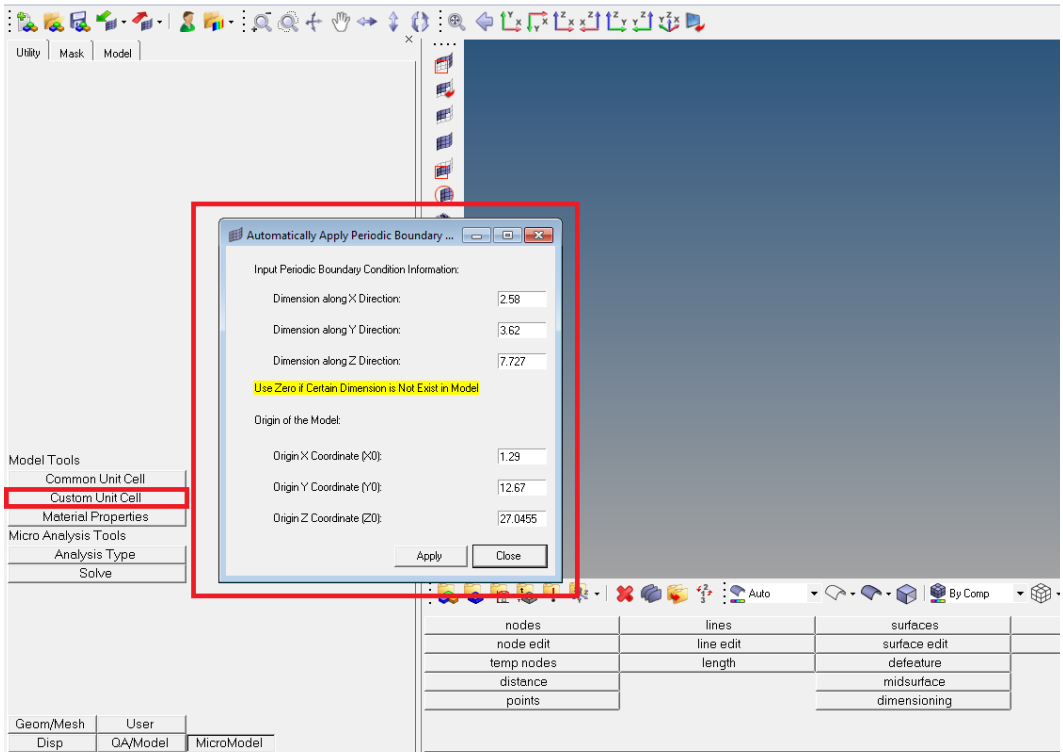

Fig. A.4: Sketch of HyperWorks-SwiftComp Micromechanics User Interface 4

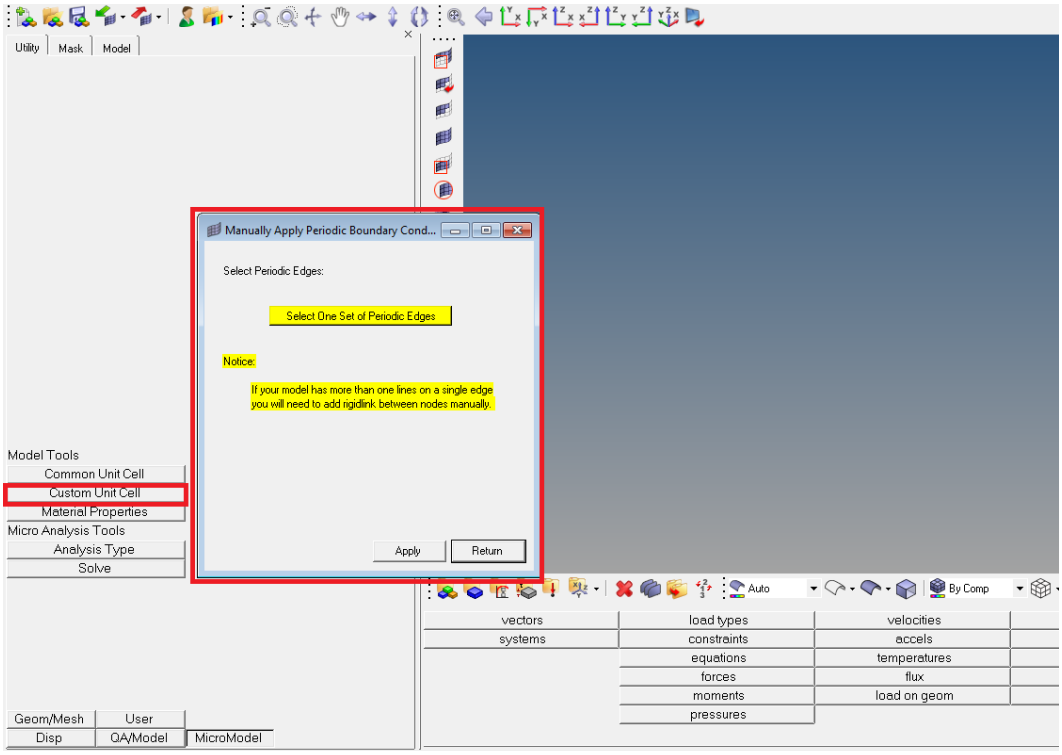

Fig. A.5: Sketch of HyperWorks-SwiftComp Micromechanics User Interface 5

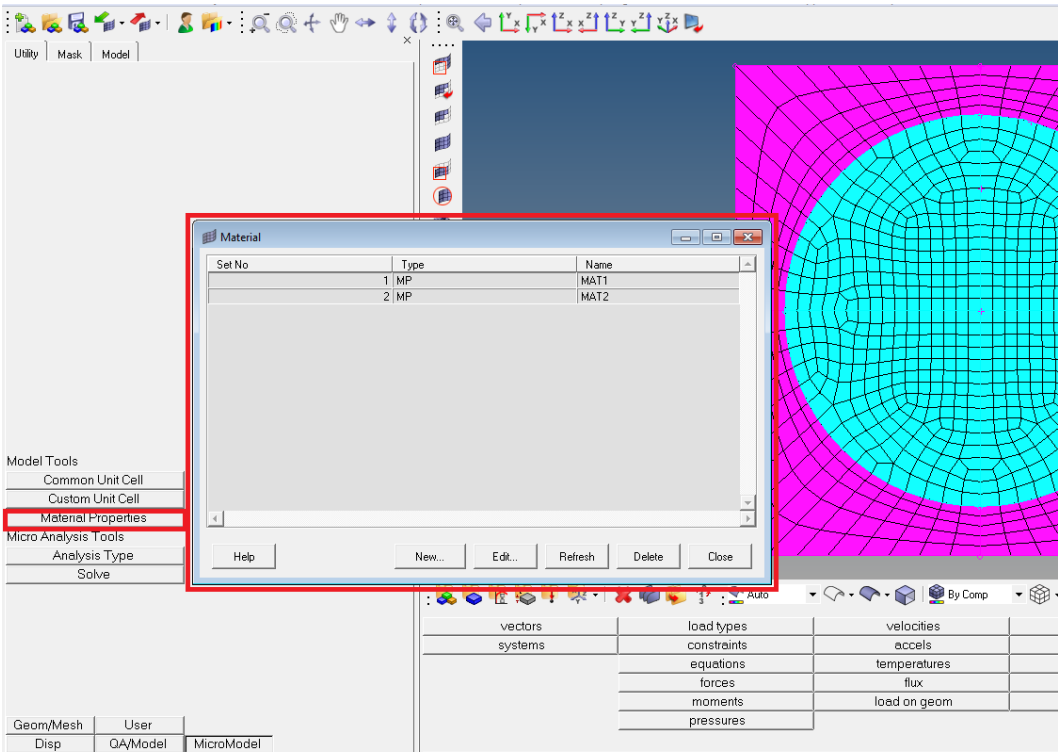

Fig. A.6: Sketch of HyperWorks-SwiftComp Micromechanics User Interface 6

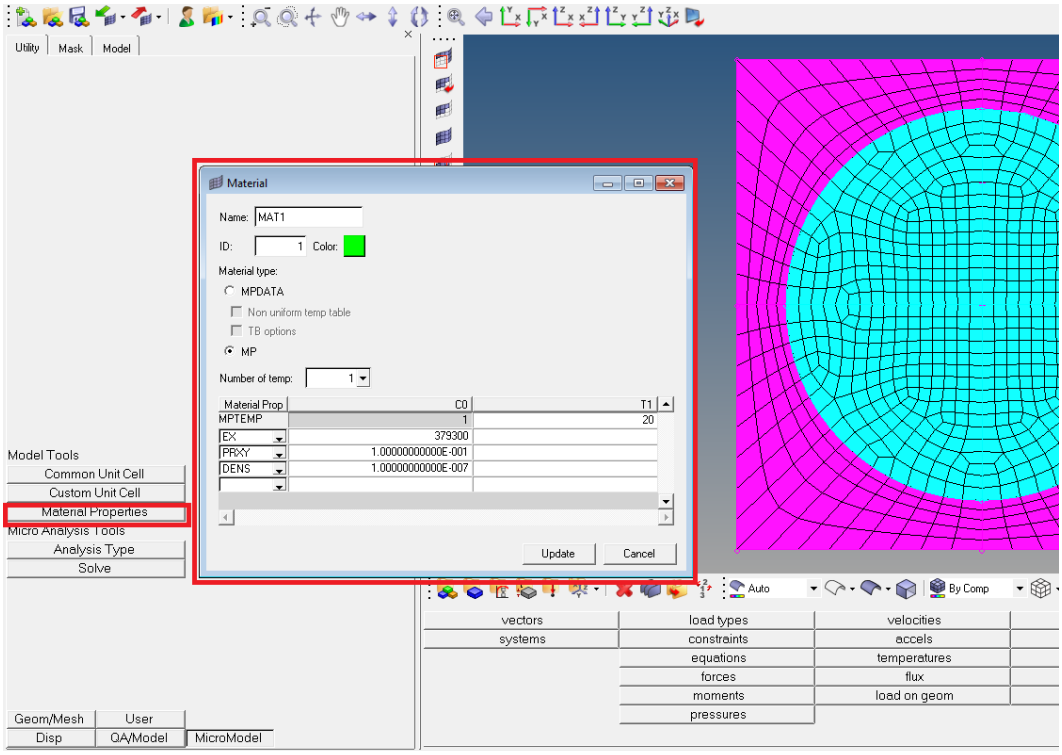

Fig. A.7: Sketch of HyperWorks-SwiftComp Micromechanics User Interface 7

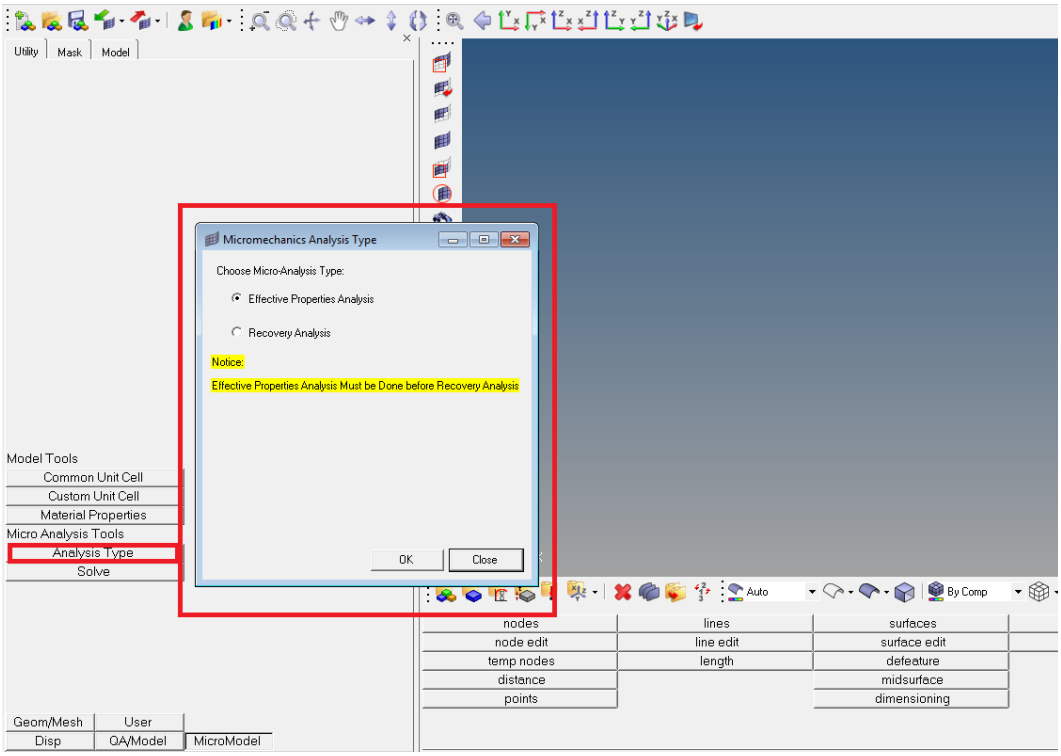

Fig. A.8: Sketch of HyperWorks-SwiftComp Micromechanics User Interface 8

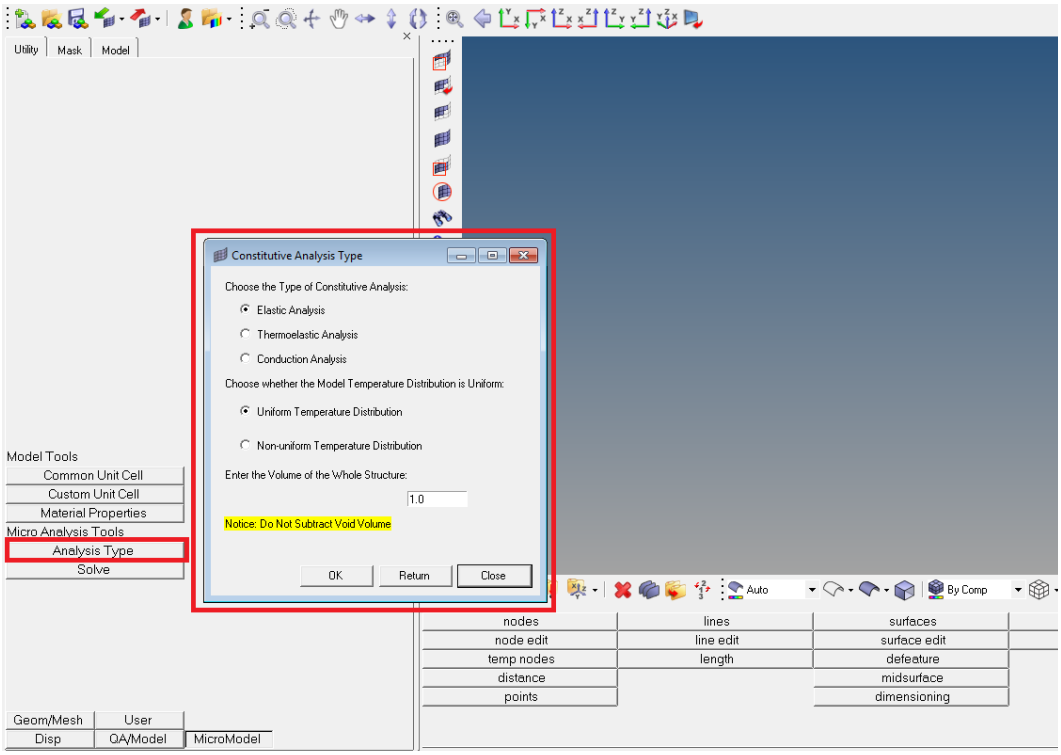

Fig. A.9: Sketch of HyperWorks-SwiftComp Micromechanics User Interface 9

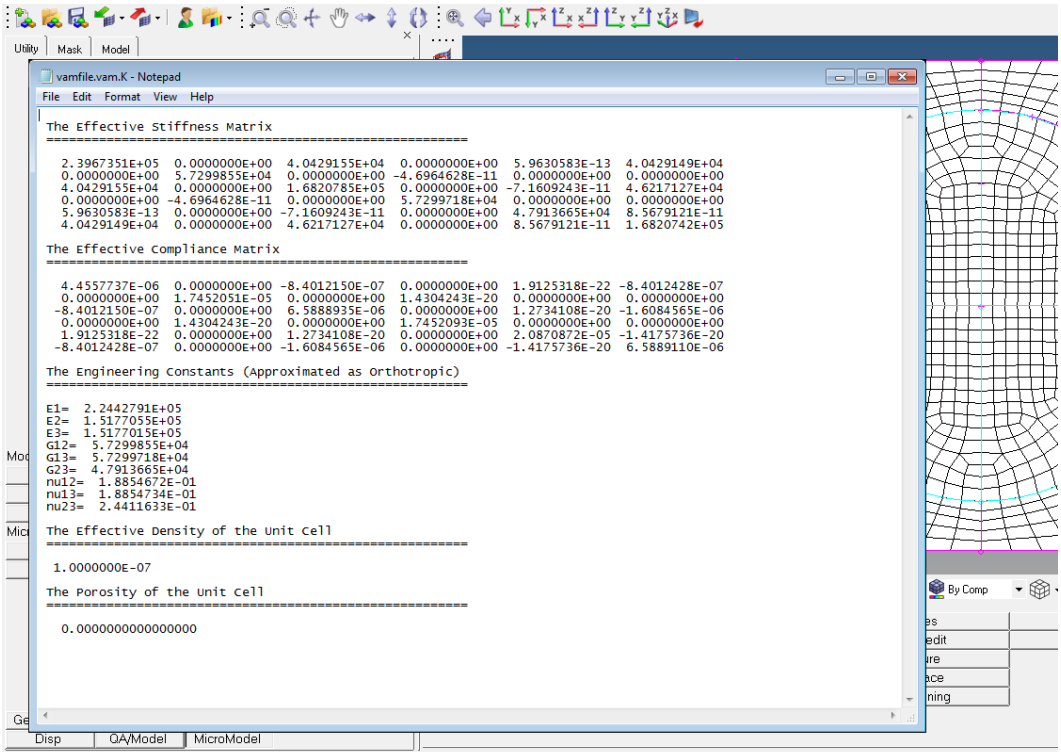

Fig. A.10: Sketch of HyperWorks-SwiftComp Micromechanics User Interface 10

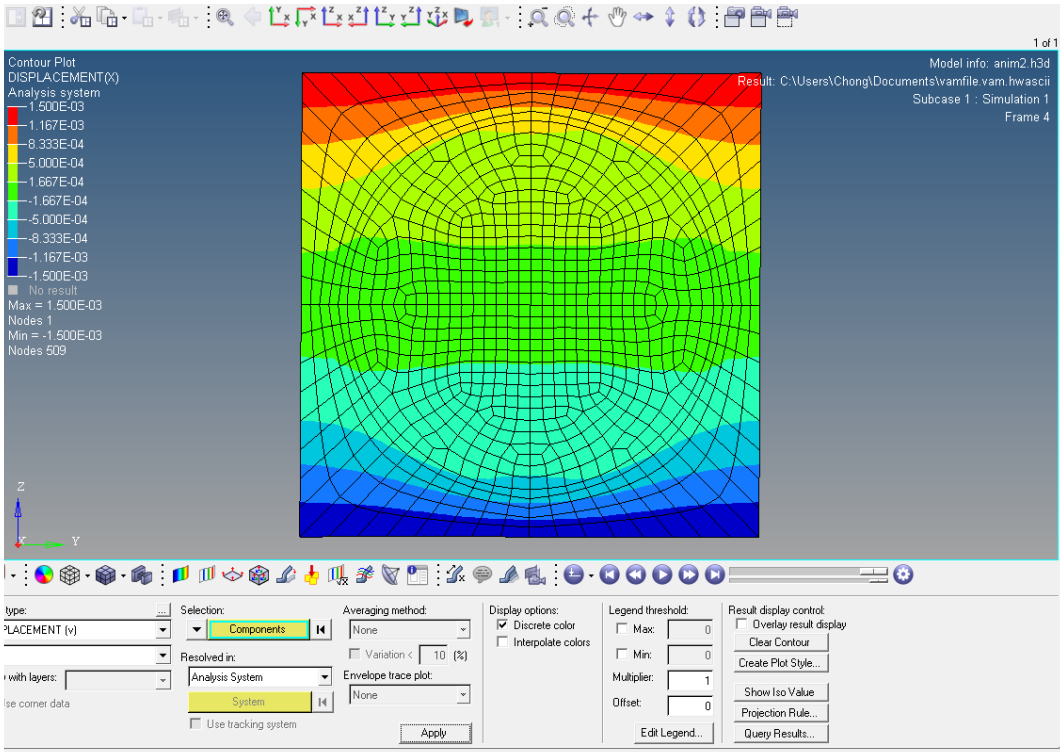

Fig. A.11: Sketch of HyperWorks-SwiftComp Micromechanics User Interface 11

tively. Under *microtype.tcl* file, the *const_analy.tpl* is included for generating the input files and *attrib.lst* gives definition of each parameter which is also borrowed from ANSYS-HyperWorks interface. The coding structure can be expressed into flow chart as shown in Fig. A.12.

A.4 The Source Code of *hm_micro* Macro File

```
*includemacrofile("globalpage.mac")
//      Page Definitions
*includemacrofile("disppage.mac")
*includemacrofile("geommeshpage.mac")
*includemacrofile("qamodelpage.mac")
*includemacrofile("userpage.mac")
*includemacrofile("micro.mac")
```

A.5 The Source Code of *globalpage* Macro File

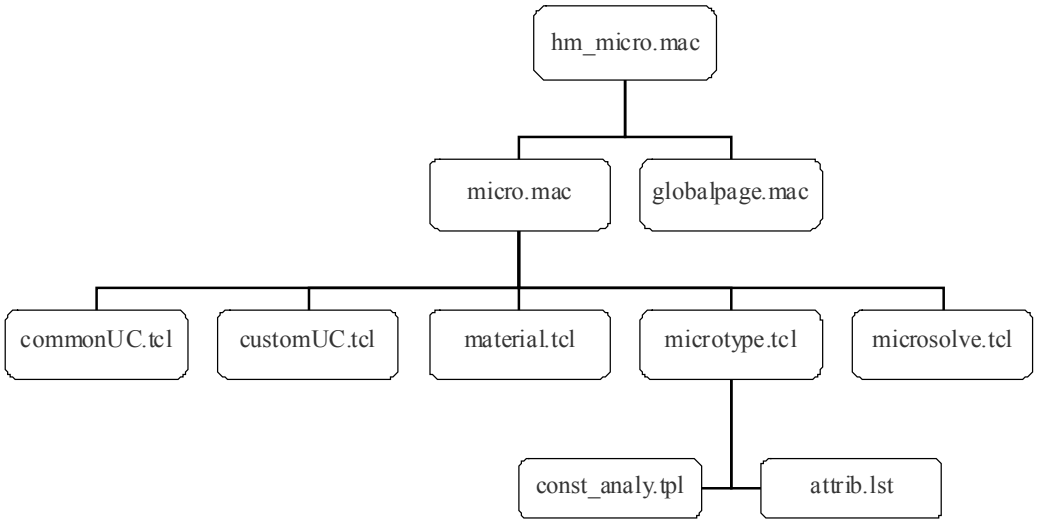

Fig. A.12: Flow chart of interface coding structure

```
*createbuttongroup(0, 0, "Disp", 1, 0, 5, BUTTON, "Display options.", "
    macroSetActivePage", 1)
*createbuttongroup(0, 0, "QA/Model", 1, 5, 5, BUTTON, "Element quality
    checking + Model Setup.", "macroSetActivePage", 2)
*createbuttongroup(0, 0, "Geom/Mesh", 2, 0, 5, BUTTON, "Geometry/Meshing
    editing tools.", "macroSetActivePage", 3)
*createbuttongroup(0, 0, "User", 2, 5, 5, BUTTON, "User defined tools.", "
    macroSetActivePage", 5)
*createbuttongroup(0, 0, "MicroModel",1, 10, 5, BUTTON, "Micromechanics
    modeling.", "macroSetActivePage", 6)
*setactivegroup(0,0,1)
*setbuttongroupactivecolor(GREY)
*beginmacro("EvalTcl")
    *evaltclscript($1,0)
*endmacro()
*beginmacro(macroSetActivePage)
   *setactivepage($1)
```

```
*endmacro()
*beginmacro(macroMacroMenuStatus)
    *enablemacromenu(0)
*endmacro()
```

A.6 The Source Code of *micro* Macro File

```
////////////////////////////////////////////////////////////////
// Filename: micro.mac
// Purpose:  Macro menu 'Micromechanics modeling' page
// definitions.
// Function: VAMUCH & HyperWorks Interface
// Copyright (C) 2013 by Wenbin Yu, and Chong Teng.
// Support: Chong Teng <chongteng@aggiemail.usu.edu>
////////////////////////////////////////////////////////////////
*createbutton(6,"Solve",10,0,10,BUTTON,"Run Micro Analysis ", "EvalTcl","
    microsolve.tcl")
*createbutton(6,"Analysis Type",11,0,10,BUTTON,"Choose Micro Analysis Type",
     "EvalTcl","microtype.tcl")
*createtext(6, "Micro Analysis Tools", 0, 0)
*createbutton(6, "Material Properties",13,0,10,BUTTON, "Define Material
    Tables", "EvalTcl","material.tcl")
*createbutton(6,"Custom Unit Cell",14,0,10,BUTTON,"User Customized Unit Cell
    ", "EvalTcl","customUC.tcl")
*createbutton(6,"Common Unit Cell",15,0,10,BUTTON,"Common Used Unit Cell", "
    EvalTcl","commonUC.tcl")
*createtext(6, "Model Tools", 0, 0)
//      User Page Macro Definitions
*beginmacro("ConnectToDebugger")
        *evaltclstring("source /Program\ Files/tclPro1.4/win32-ix86/bin/
            prodebug.tcl; debugger_init;",0)
*endmacro()
*beginmacro("LaunchWidgetTour")
        //  Purpose:    Launch the HyperWorks Widget Tour dialog.
        *evaltclstring("::VAMUCHUI::WidgetTour",0)
```

```
*endmacro()
*beginmacro("EvalTcl")
   *evaltclscript($1,0)
*endmacro()
```

A.7 The Source Code of *commonUC* Tcl File

```
#######################################################################
## Filename: commonUC.tcl
## Purpose:  Build up micromechanics models
## Function: VAMUCH & HyperWorks Interface
## Copyright (C) 2013 by Wenbin Yu, and Chong Teng.
## Support: Chong Teng <chongteng@aggiemail.usu.edu>
##
#######################################################################



namespace eval ::altair::commonUC \
{
    variable recess;
    variable fillType;
    variable namemark;
    variable altair_dir;

    array set VoF \
    {
        present          0
        past             0
    }
    array set LoS \
    {
        present          0
        past             0
    }
```

```
array set EX1 \
{
    present          0
    past             0
}
array set PRXY1 \
{
    present          0
    past             0
}
array set EX2 \
{
    present          0
    past             0
}
array set PRXY2 \
{
    present          0
    past             0
}
array set DENS1 \
{
    present          0
    past             0
}
array set DENS2 \
{
    present          0
    past             0
}
array set TREF \
{
    present          0
    past             0
}
```

```
    array set meshpara\
    {
       1                0
       2                0
       3                0
       4                0
       5                0
       6                0
       7                0
       8                0
       9                0
    }
}

set altair_dir [hm_info -appinfo ALTAIR_HOME];
*templatefileset "$altair_dir/templates/feoutput/ansys/ansys.tpl"


namespace eval ::altair::ucparam \
{
    variable recess;
    variable namemark;
}



proc ::altair::commonUC::commonUCdialog {} {

    if {[winfo exists .cmonUCopt]} {
        destroy .cmonUCopt;
    }

    variable    recess;
```

```
    set miny 150
    if {![::hwt::OnPc]} {set miny 175};
#####################################################################
## create window and buttons
#####################################################################
    ::hwt::CreateWindow cmonUCopt \
        -windowtitle   "Common Unit Cell Options" \
        -cancelButton   "Close" \
        -cancelFunc     ::altair::commonUC::Quit \
        -addButton  Select ::altair::commonUC::Select no_icon \
        -minsize 350 $miny  \
        post;
    set xloc [ ::fepre::GetXLocation 350 ];
    set yloc [ ::fepre::GetYLocation 190 ];
    if {$yloc < 0} {set yloc 100};   ##yloc coming out negative on linux
        sometimes
    wm geometry .cmonUCopt +$xloc+$yloc;

    KeepOnTop   .cmonUCopt

    set recess [ ::hwt::WindowRecess cmonUCopt ];

    grid columnconfigure $recess 1 -weight 3

    set ::altair::commonUC::fillType 2Dstand;

#####################################################################
## define option layouts
#####################################################################

    radiobutton $recess.2DstandUC \
        -text "2D Standard Unit Cell" \
        -variable ::altair::commonUC::fillType \
        -value 2Dstand \
```

```
        -state normal;

    radiobutton $recess.3DstandUC \
        -text "3D Standard Unit Cell" \
        -variable ::altair::commonUC::fillType \
        -value 3Dstand \
        -state normal;

  #  radiobutton $recess.2DrandUC \
  #      -text "2D Random Unit Cell" \
  #      -variable ::altair::commonUC::fillType \
  #      -value 2Drand \
  #      -state normal;

  #  radiobutton $recess.3DrandUC \
  #      -text "3D Random Unit Cell" \
  #      -variable ::altair::commonUC::fillType \
  #      -value 3Drand \
  #      -state normal;

    label $recess.l1\
       -text "Choose the Type of Common Unit Cell:" \
       -state normal;

#####################################################################
## position of the options
#####################################################################
   grid $recess.l1     -row 1 -column 0 -padx 20 -pady 15 -sticky nw;
    grid $recess.2DstandUC  -row 2 -column 0 -padx 20 -pady 10 -sticky nw;
    grid $recess.3DstandUC  -row 3 -column 0 -padx 20 -pady 10 -sticky nw;
#    grid $recess.2DrandUC  -row 4 -column 0 -padx 20 -pady 10 -#sticky nw;
#    grid $recess.3DrandUC  -row 5 -column 0 -padx 20 -pady 10 -#sticky nw;


    ::hwt::RemoveDefaultButtonBinding $recess
```

```
};


proc ::altair::commonUC::Quit {} \
{
    ::hwt::UnpostWindow cmonUCopt
    hm_usermessage "";
};

#Call the function
::altair::commonUC::commonUCdialog


# parent function called by triggering the select button.
proc ::altair::commonUC::Select {} \
{
    if {[winfo exists .ucparam]} {
        destroy .ucparam;
    }

    variable fillType;
    variable namemark;
    variable recess;
    variable VoF;
    variable LoS;
    variable EX1;
    variable PRXY1;
    variable EX2;
    variable PRXY2;



# get the name of the new window

if { $fillType == "2Dstand" } \
```

```
{
 set namemark "2D Standard Unit Cell" ;
} \
elseif { $fillType == "3Dstand" } \
{
 set namemark "3D Standard Unit Cell" ;
} \
elseif { $fillType == "2Drand" } \
{
 set namemark "2D Random Unit Cell" ;
} \
else \
{
 set namemark "3D Random Unit Cell" ;
}


    ::altair::commonUC::Quit;
    ::hwt::CreateWindow ucparam \
       -windowtitle  "$namemark"\
       -cancelButton   "Close" \
       -cancelFunc     ::altair::ucparam::Quit \
       -addButton Create ::altair::commonUC::Create no_icon \
       -addButton Return ::altair::ucparam::Return no_icon \
       -minsize 350 150  \
        post;
    set xloc [ ::fepre::GetXLocation 350 ];
    set yloc [ ::fepre::GetYLocation 190 ];
    if {$yloc < 0} {set yloc 100};   ##yloc coming out negative on linux
        sometimes
    wm geometry .ucparam +$xloc+$yloc;

    KeepOnTop   .ucparam

    set recess [ ::hwt::WindowRecess ucparam ];
```

```
    grid columnconfigure $recess 1 -weight 3

    label $recess.l2 -text "Input $namemark Information:" -state normal;

if { $fillType == "2Dstand" } \
{
    label $recess.l3 -text "Volume Fraction(0.5~75.0):" -state normal;
}\
elseif { $fillType == "3Dstand" } \
{
    label $recess.l3 -text "Volume Fraction(0.1~49.0):" -state normal;
};
    label $recess.l4 -text "Length of Square:" -state normal;
    label $recess.l5 -text "Material 1, EX, PRXY, DENS:" -state normal;
    label $recess.l6 -text "Material 2, EX, PRXY, DENS:" -state normal;
    label $recess.l7 -text "Reference Temperature:" -state normal;

    entry $recess.e1 \
            -textvariable ::altair::commonUC::VoF(present) \
            -width 10 -justify left \
            -state normal;
        set ::altair::commonUC::VoF(present) 40;
    entry $recess.e2 \
            -textvariable ::altair::commonUC::LoS(present) \
            -width 10 -justify left \
            -state normal;
        set ::altair::commonUC::LoS(present) 1;
    entry $recess.e3 \
            -textvariable ::altair::commonUC::EX1(present) \
            -width 10 -justify left \
            -state normal;
        set ::altair::commonUC::EX1(present) 379300;
    entry $recess.e4 \
            -textvariable ::altair::commonUC::PRXY1(present) \
```

```
            -width 5 -justify left \
            -state normal;
        set ::altair::commonUC::PRXY1(present) 0.1;
    entry $recess.e5 \
            -textvariable ::altair::commonUC::EX2(present) \
            -width 10 -justify left \
            -state normal;
        set ::altair::commonUC::EX2(present) 68300;
    entry $recess.e6 \
            -textvariable ::altair::commonUC::PRXY2(present) \
            -width 5 -justify left \
            -state normal;
        set ::altair::commonUC::PRXY2(present) 0.3;

    entry $recess.e7 \
            -textvariable ::altair::commonUC::DENS1(present) \
            -width 10 -justify left \
            -state normal;
        set ::altair::commonUC::DENS1(present) 1E-7;
    entry $recess.e8 \
            -textvariable ::altair::commonUC::DENS2(present) \
            -width 10 -justify left \
            -state normal;
        set ::altair::commonUC::DENS2(present) 1E-7;
    entry $recess.e9 \
            -textvariable ::altair::commonUC::TREF(present) \
            -width 10 -justify left \
            -state normal;
        set ::altair::commonUC::TREF(present) 20;

grid $recess.l2     -row 1 -column 0 -padx 0 -pady 15 -sticky nw;
grid $recess.l3     -row 2 -column 0 -padx 0 -pady 8 -sticky nw;
grid $recess.e1     -row 2 -column 1 -pady 8 -sticky nw;
grid $recess.l4     -row 3 -column 0 -padx 0 -pady 8 -sticky nw;
grid $recess.e2     -row 3 -column 1 -pady 8 -sticky nw;
```

```
grid $recess.l5     -row 4 -column 0 -padx 0 -pady 8 -sticky nw;
grid $recess.e3     -row 4 -column 1 -pady 8 -sticky nw;
grid $recess.e4     -row 4 -column 2 -pady 8 -sticky nw;
grid $recess.e7     -row 4 -column 3 -pady 8 -sticky nw;
grid $recess.l6     -row 5 -column 0 -padx 0 -pady 8 -sticky nw;
grid $recess.e5     -row 5 -column 1 -pady 8 -sticky nw;
grid $recess.e6     -row 5 -column 2 -pady 8 -sticky nw;
grid $recess.e8     -row 5 -column 3 -pady 8 -sticky nw;
grid $recess.l7     -row 6 -column 0 -padx 0 -pady 8 -sticky nw;
grid $recess.e9     -row 6 -column 1 -pady 8 -sticky nw;
};


proc ::altair::ucparam::Quit {} \
{
    ::hwt::UnpostWindow ucparam;
};

proc ::altair::ucparam::Return {} \
{
    ::hwt::UnpostWindow ucparam;
    ::hwt::PostWindow cmonUCopt;
};

proc ::altair::commonUC::Create {} \
{
    variable fillType;
    variable namemark;
    variable recess;
    variable VoF;
    variable LoS;
    variable EX1;
    variable PRXY1;
    variable EX2;
    variable PRXY2;
```

```
    variable radius;
    variable Pi;
    variable length;
    variable blcc;
    variable blcl;
    variable countnum;
    variable meshpara;
    variable miniblcl;
    variable DENS1;
    variable DENS2;
    variable TREF;

set Pi [expr 2*asin(1.0)];
set blcc [expr $LoS(present)/4.0000];
set blcl [expr $LoS(present)/2.0000];

if { $fillType == "2Dstand" } \
{
set radius [expr sqrt($VoF(present)*$LoS(present)**2.0000/($Pi*100.0000))];

#create 1/4 unit cell

*surfacemode 4
*createplane 1 1.0000 0.0000 0.0000 0.0000 $blcc $blcc
*surfaceplane 1 $blcl
*createnode 0 0 0 0 0 0
*createlist nodes 1  1
*createvector 1 1.0000 0.0000 0.0000
*createcirclefromcenterradius 1 1 $radius 360 0
*clearlist nodes 1  1
*createmark lines 1  5
*linesplitatline 1 3
*createmark lines 1  6
*linesplitatline 1 4
*createmark lines 1  9 7
```

```
*deletemark lines 1
*createmark lines 1 "all"
*renumber lines 1 1 1 0 0

*createmark surfaces 1  1
*createmark lines 2  2
*createvector 1 1 0000 0.0000 0.0000
*surfacemarksplitwithlines 1 2 1 1 0
*normalsoff

*createmark nodes 1 1-3
*nodemarkcleartempmark 1



#create 1/4 mesh



*setedgedensitylink 0
*elementorder 1
*createmark surfaces 1  1
*interactiveremeshsurf 1 0.10001 2 2 2 1 1
*set_meshfaceparams 0 5 2 0 0 1 0.5 1 1
*automesh 0 5 2
*set_meshfaceparams 0 4 1 0 0 1 0.5 1 1
*set_meshedgeparams 0 8 1 0 0 0 0.10001 0 0
*set_meshedgeparams 1 5 1 0 0 0 0.10001 0 0
*set_meshedgeparams 2 16 1 0 0 0 0.10001 0 0
*set_meshedgeparams 3 5 1 0 0 0 0.10001 0 0
*set_meshedgeparams 4 8 1 0 0 0 0.10001 0 0
*automesh 0 4 1
*storemeshtodatabase 1
*ameshclearsurface
```

```
*setedgedensitylink 0
*elementorder 1
*createmark surfaces 1  2
*interactiveremeshsurf 1 0.10001 2 2 2 1 1
*set_meshfaceparams 0 3 2 0 0 1 0.5 1 1
*automesh 0 5 1
*set_meshfaceparams 0 4 1 0 0 1 0.5 1 1
*set_meshedgeparams 0 16 1 0 0 0 0.10001 0 0
*set_meshedgeparams 2 16 1 0 0 0 0.10001 0 0
*automesh 0 5 1
*storemeshtodatabase 1
*ameshclearsurface


#create 2nd surface by reflecting with y axis and reflect

*createmark surfaces 1  1 2
*duplicatemark surfaces 1 1
*createmark surfaces 1  3 4
*createvector 1 0.0000 1.0000 0.0000
*translatemark surfaces 1 1 -$blcl
*createmark surfaces 1  3 4
*createplane 1 0.0000 1.0000 0.0000 0.0000 -$blcc $blcc
*reflectmark surfaces 1 1


#create 3rd and 4th surfaces and reflect

*createmark surfaces 1  1-4
*duplicatemark surfaces 1 1
*createmark surfaces 1  5-8
*createvector 1 0.0000 0.0000 1.0000
*translatemark surfaces 1 1 -$blcl
*createmark surfaces 1  5-8
*createplane 1 0.0000 0.0000 1.0000 0.0000 0.0000 -$blcc
```

```
*reflectmark surfaces 1 1


#duplicate mesh and reflect

##calculate element numbers

set countnum(1) [hm_count elements all 0 0]
set countnum(2) [expr $countnum(1)+1]
set countnum(3) [expr $countnum(1)*2]
set countnum(4) [expr $countnum(3)+1]
set countnum(5) [expr $countnum(3)*2]

##start duplicate and reflect

*createmark elements 1  1-$countnum(1)
*duplicatemark elements 1 1
*createmark elements 1  $countnum(2)-$countnum(3)
*createvector 1 0.0000 1.0000 0.0000
*translatemark elements 1 1 -$blcl
*createmark elements 1  $countnum(2)-$countnum(3)
*createplane 1 0.0000 1.0000 0.0000 0.0000 -$blcc $blcc
*reflectmark elements 1 1
*createmark elements 1 1-$countnum(3)
*duplicatemark elements 1 1
*createmark elements 1  $countnum(4)-$countnum(5)
*createvector 1 0.0000 0.0000 1.0000
*translatemark elements 1 1 -$blcl
*createmark elements 1  $countnum(4)-$countnum(5)
*createplane 1 0.0000 0.0000 1.0000 0.0000 0.0000 -$blcc
*reflectmark elements 1 1


#create addtional component collecter
```

```
*createmark materials 1
*clearmark materials 1
*collectorcreateonly components "fiber" "" 7
*createmark components 1
*clearmark components 1
*createmark components 1  "fiber"
*materialupdate components 1 ""
*createmark components 1
*clearmark components 1
*createmark components 1  "fiber"
*clearmark components 1
*createmark properties 1
*clearmark properties 1
*createmark materials 1
*clearmark materials 1
*createmark elements 1
*clearmark elements 1

#organize surface

*createmark surfaces 1  2 4 6 8
*movemark surfaces 1 "fiber"
*retainmarkselections 1
*renamecollector components "auto1" "matrix"
*retainmarkselections 0
*retainmarkselections 1
*createmark components 1  "matrix"
*colormark components 1 8
*retainmarkselections 0


#organize elements
set countnum(6) [expr $countnum(1)+81]
set countnum(7) [expr $countnum(3)+81]
set countnum(8) [expr $countnum(1)*3]
```

```
set countnum(9) [expr $countnum(8)+81]

*createmark elements 1  81-$countnum(1) $countnum(6)-$countnum(3) $countnum
    (7)-$countnum(8) $countnum(9)-$countnum(5)
*movemark elements 1 "fiber"

#merge edge nodes
*createmark elements 1  "all"
*equivalence elements 1 1e-006 1 0 0



#renumber nodes and elements
*renumbersolveridall 1 1 0 0 0 0 0
*createmark nodes 2 "all"
*renumber nodes 2 1 1 0 0
*createmark elements 1 "all"
*renumber elements 1 1 1 0 0

#display control
*window 0 0 0 0 0
*view "rear"
*setdisplayattributes 2 0

#create material & properties (follow Ansys template)



*collectorcreate materials "MAT1" "" 4
*createmark materials 1  "MAT1"
*renumber materials 1 1 1 0 0
*createmark materials 1  "MAT1"
*dictionaryload materials 1 "C:/Program Files/Altair/11.0/templates/feoutput
    /ansys/ansys.tpl" "MATERIAL"
*attributeupdateint materials 1 504 8 0 0 1
*createdoublearray 1  $TREF(present)
*attributeupdatedoublearray materials 1 505 8 2 0 1 1
```

```
*attributeupdateint materials 1 502 8 2 0 1
*attributeupdateint materials 1 56 8 0 0 1
*attributeupdateint materials 1 2644 8 0 0 1
*createdoublearray 1  $EX1(present)
*attributeupdatedoublearray materials 1 2645 8 2 0 1 1
*attributeupdateint materials 1 76 8 0 0 1
*attributeupdateint materials 1 2571 8 0 0 1
*createdoublearray 1  $PRXY1(present)
*attributeupdatedoublearray materials 1 2611 8 2 0 1 1
*attributeupdateint materials 1 502 8 2 0 1
*attributeupdateint materials 1 53 8 0 0 1
*attributeupdateint materials 1 2579 8 0 0 1
*createdoublearray 1 $DENS1(present)
*attributeupdatedoublearray materials 1 2619 8 2 0 1 1



*collectorcreate materials "MAT2" "" 3
*createmark materials 1  "MAT2"
*renumber materials 1 2 1 0 0
*createmark materials 1  "MAT2"
*dictionaryload materials 1 "C:/Program Files/Altair/11.0/templates/feoutput
    /ansys/ansys.tpl" "MATERIAL"
*attributeupdateint materials 2 504 8 0 0 1
*createdoublearray 1  $TREF(present)
*attributeupdatedoublearray materials 2 505 8 2 0 1 1
*attributeupdateint materials 2 502 8 2 0 1
*attributeupdateint materials 2 56 8 0 0 1
*attributeupdateint materials 2 2644 8 0 0 1
*createdoublearray 1  $EX2(present)
*attributeupdatedoublearray materials 2 2645 8 2 0 1 1
*attributeupdateint materials 2 76 8 0 0 1
*attributeupdateint materials 2 2571 8 0 0 1
*createdoublearray 1  $PRXY2(present)
*attributeupdatedoublearray materials 2 2611 8 2 0 1 1
```

```
*attributeupdateint materials 2 502 8 2 0 1
*attributeupdateint materials 2 53 8 0 0 1
*attributeupdateint materials 2 2579 8 0 0 1
*createdoublearray 1 $DENS2(present)
*attributeupdatedoublearray materials 2 2619 8 2 0 1 1



*collectorcreateonly properties "PROP1" "" 5
*collectorcreateonly properties "PROP2" "" 6

#assign materials & properties to components

*createmark components 1  "fiber"
*materialupdate components 1 "MAT1"
*clearmark components 1
*createmark components 1  "fiber"
*propertyupdate components 1 "PROP1"
*clearmark components 1
*createmark components 1  "matrix"
*materialupdate components 1 "MAT2"
*clearmark components 1
*createmark components 1  "matrix"
*propertyupdate components 1 "PROP2"
*clearmark components 1



}\
elseif { $fillType == "3Dstand" } \
{
set radius [expr (0.0300*$VoF(present)*$LoS(present)**3.0000/(4.0000*$Pi))
    **(1.0000/3.0000)];

#create 1/8 unit cell
```

```
*solidblock 0 0 $blcl 0 -$blcl 0 0 0 -$blcl $blcl 0 0
*createnode 0 0 0 0 0 0
*createnode 0 -$blcl 0 0 0 0
*createnode $blcl 0 0 0 0 0
*surfacemode 4
*surfacespherefromthreepoints 1 $radius 3 0 90 2 0 -90 0
*createmark solids 1  1
*createmark lines 1  13-15
*body_split_with_lines solids 1 1 0
*createmark materials 1
*clearmark materials 1
*collectorcreateonly components "particle" "" 7
*createmark components 1
*clearmark components 1
*createmark components 1  "particle"
*materialupdate components 1 ""
*createmark components 1
*clearmark components 1
*createmark components 1  "particle"
*clearmark components 1
*createmark properties 1
*clearmark properties 1
*createmark materials 1
*clearmark materials 1
*createmark elements 1
*clearmark elements 1

*createmark solids 1  1
*movemark solids 1 "auto1"
*createmark solids 1  2
*movemark solids 1 "particle"

*retainmarkselections 1
*createmark components 1  "auto1"
```

```
*colormark components 1 8
*renamecollector components "auto1" "matrix"
*retainmarkselections 0

##delete 3 nodes that used for generalize partial sphere
*createmark nodes 1  1-3
*nodemarkcleartempmark 1

#create 1/8 mesh

##create a small block particle for spherical particle

set miniblcl [expr 0.3000*$radius];

*linecreatestraight $miniblcl -$blcl 0 $miniblcl 0 0

*createmark solids 1 "by comp" "particle"
*createdoublearray 3  0 0 1
*createlist lines 1  "-1"
*body_split_with_morphed_lines solids 1 1 1 3 0 1

*linecreatestraight 0 0 $miniblcl $blcl 0 $miniblcl

*createmark solids 1  "-1"
*createdoublearray 3  0 1 0
*createlist lines 1  "-1"
*body_split_with_morphed_lines solids 1 1 1 3 0 1

*linecreatestraight 0 -$miniblcl 0 0 -$miniblcl $blcl

*createmark solids 1  "-1"
*createdoublearray 3  1 0 0
*createlist lines 1  "-1"
*body_split_with_morphed_lines solids 1 1 1 3 0 1
```

```
##find the surfaces that needs to be merged

*createmark solids 1  2
*findmark solids 1 1 1 surfaces 0 2
set surf1 [hm_getmark surfaces 2]
*createmark solids 1  3
*findmark solids 1 1 1 surfaces 0 2
set surf2 [hm_getmark surfaces 2]
set intersect1 [list]
foreach elem $surf1\
{
   if {$elem in $surf2}\
   {
      lappend intersect1 $elem;
    }
}

*createmark solids 1  2
*findmark solids 1 1 1 surfaces 0 2
set surf1 [hm_getmark surfaces 2]
*createmark solids 1  5
*findmark solids 1 1 1 surfaces 0 2
set surf2 [hm_getmark surfaces 2]
set intersect2 [list]
foreach elem $surf1\
{
   if {$elem in $surf2}\
   {
      lappend intersect2 $elem;
    }
}

*createmark solids 1  3
*findmark solids 1 1 1 surfaces 0 2
```

```
set surf1 [hm_getmark surfaces 2]
*createmark solids 1  5
*findmark solids 1 1 1 surfaces 0 2
set surf2 [hm_getmark surfaces 2]
set intersect3 [list]
foreach elem $surf1\
{
   if {$elem in $surf2}\
   {
      lappend intersect3 $elem;
    }
}

##merge solids and delete cut lines

*createmark surfaces 1 $intersect1 $intersect2 $intersect3
*solid_untrim 1 0
*renumbersolveridall 1 1 0 0 0 0 0
*createmark lines 1  1-3
*deletemark lines 1

##create mesh for matrix

*setedgedensitylink 0
*elementorder 1
*createmark surfaces 1  2
*interactiveremeshsurf 1 10 2 2 2 1 1
*set_meshfaceparams 0 2 2 0 0 1 0.5 1 1
*automesh 0 2 2
*set_meshfaceparams 0 4 1 0 0 1 0.5 1 1
*set_meshedgeparams 0 4 1 0 0 0 10 0 0
*set_meshedgeparams 1 4 1 0 0 0 10 0 0
*set_meshedgeparams 2 4 1 0 0 0 10 0 0
*set_meshedgeparams 3 4 1 0 0 0 10 0 0
*automesh 0 4 2
```

```
*storemeshtodatabase 0
*ameshclearsurface

*setedgedensitylink 0
*elementorder 1
*createmark surfaces 1  1
*interactiveremeshsurf 1 10 2 2 2 1 1
*set_meshfaceparams 0 5 2 0 0 1 0.5 1 1
*automesh 0 5 2
*set_meshfaceparams 0 4 2 0 0 1 0.5 1 1
*set_meshedgeparams 0 4 1 0 0 0 10 0 0
*set_meshedgeparams 2 2 1 0 0 0 10 0 0
*set_meshedgeparams 3 8 1 0 0 0 10 0 0
*set_meshedgeparams 4 2 1 0 0 0 10 0 0
*automesh 0 4 2
*storemeshtodatabase 0
*ameshclearsurface

*setedgedensitylink 0
*elementorder 1
*createmark surfaces 1  3
*interactiveremeshsurf 1 10 2 2 2 1 1
*set_meshfaceparams 0 5 2 0 0 1 0.5 1 1
*automesh 0 5 2
*set_meshfaceparams 0 4 2 0 0 1 0.5 1 1
*set_meshedgeparams 0 4 1 0 0 0 10 0 0
*set_meshedgeparams 1 2 1 0 0 0 10 0 0
*set_meshedgeparams 2 8 1 0 0 0 10 0 0
*automesh 0 4 2
*storemeshtodatabase 0
*ameshclearsurface

*setedgedensitylink 0
*elementorder 1
*createmark surfaces 1  5
```

```
*interactiveremeshsurf 1 10 2 2 2 1 1
*set_meshfaceparams 0 5 2 0 0 1 0.5 1 1
*automesh 0 5 2
*set_meshfaceparams 0 4 2 0 0 1 0.5 1 1
*set_meshedgeparams 0 4 1 0 0 0 10 0 0
*set_meshedgeparams 1 4 1 0 0 0 10 0 0
*set_meshedgeparams 3 8 1 0 0 0 10 0 0
*automesh 0 4 2
*storemeshtodatabase 0
*ameshclearsurface

*setedgedensitylink 0
*elementorder 1
*createmark surfaces 1  4
*interactiveremeshsurf 1 10 2 2 2 1 1
*set_meshfaceparams 0 2 2 0 0 1 0.5 1 1
*automesh 0 2 2
*set_meshfaceparams 0 4 2 0 0 1 0.5 1 1
*set_meshedgeparams 1 4 1 0 0 0 10 0 0
*automesh 0 4 2
*storemeshtodatabase 0
*ameshclearsurface

*setedgedensitylink 0
*elementorder 1
*createmark surfaces 1  6
*interactiveremeshsurf 1 10 2 2 2 1 1
*set_meshfaceparams 0 2 2 0 0 1 0.5 1 1
*automesh 0 2 2
*storemeshtodatabase 0
*ameshclearsurface

*createmark elements 2 "all"
*createmark solids 1  1
*solidmap_solids_begin 1 828034 0.1
```

```
  *createmark lines 1  29
  *solidmap_solids_set_elemsize 1 0.0427116493
*solidmap_solids_end
*deletemark elements 2
*createmark elements 1  "all"
*movemark elements 1 "matrix"

##mesh the particle

*solidmap_begin 0
*solidmap_prepare_usrdataptr "SOURCE" 4
*solidmap_prepare_usrdataptr "DEST" 4
*solidmap_prepare_usrdataptr "ALONG" 32
*createmark solids 1  4
*solid_prepare_entitylst solids 0
*solidmap_end 74498 4 0 0


*solidmap_begin 0
*solidmap_prepare_usrdataptr "SOURCE" 4
*solidmap_prepare_usrdataptr "DEST" 4
*solidmap_prepare_usrdataptr "ALONG" 32
*createmark solids 1  2
*solid_prepare_entitylst solids 0
*solidmap_end 74498 4 0 0

*renumbersolveridall 1 1 0 0 0 0 0


#create another 1/2 solid by reflecting with x and y axis and reflect

*createmark solids 1  "all"
*duplicatemark solids 1 1
*createmark solids 1  5 4 6
*createvector 1 1.0000 0.0000 0.0000
```

```
*translatemark solids 1 1 -$blcl
*createmark solids 1  5 4 6
*createplane 1 1.0000 0.0000 0.0000 -$blcc 0.0000 0.0000
*reflectmark solids 1 1

*createmark solids 1  "all"
*duplicatemark solids 1 1
*createmark solids 1  7-12
*createvector 1 0.0000 1.0000 0.0000
*translatemark solids 1 1 $blcl
*createmark solids 1  7-12
*createplane 1 0.0000 1.0000 0.0000 0.0000 $blcc 0.0000
*reflectmark solids 1 1

#create the rest 1/2 solid and reflect

*createmark solids 1  "all"
*duplicatemark solids 1 1
*createmark solids 1  13-24
*createvector 1 0.0000 0.0000 1.0000
*translatemark solids 1 1 -$blcl
*createmark solids 1  13-24
*createplane 1 0.0000 0.0000 1.0000 0.0000 0.0000 -$blcc
*reflectmark solids 1 1

#organize solids

*createmark solids 1  4 7 11 13 17 19 23
*movemark solids 1 "matrix"


#duplicate mesh and reflect them

#renumber nodes and elements
```

```
*createmark nodes 1 "all"
*renumber nodes 1 1 1 0 0
*createmark elements 1  "all"
*renumber elements 1 1 1 0 0

##calculate element numbers

set countnum(1) [hm_count elements all 0 0]
set countnum(2) [expr $countnum(1)+1]
set countnum(3) [expr $countnum(1)*2]
set countnum(4) [expr $countnum(3)+1]
set countnum(5) [expr $countnum(3)*2]
set countnum(6) [expr $countnum(5)+1]
set countnum(7) [expr $countnum(5)*2]

##start duplicate and reflect
*createmark elements 1 "all"
*duplicatemark elements 1 1
*createmark elements 1  $countnum(2)-$countnum(3)
*createvector 1 1.0000 0.0000 0.0000
*translatemark elements 1 1 -$blcl
*createmark elements 1  $countnum(2)-$countnum(3)
*createplane 1 1.0000 0.0000 0.0000 -$blcc 0.0000 0.0000
*reflectmark elements 1 1

*createmark elements 1 "all"
*duplicatemark elements 1 1
*createmark elements 1  $countnum(4)-$countnum(5)
*createvector 1 0.0000 1.0000 0.0000
*translatemark elements 1 1 $blcl
*createmark elements 1  $countnum(4)-$countnum(5)
*createplane 1 0.0000 1.0000 0.0000 0.0000 $blcc 0.0000
*reflectmark elements 1 1

*createmark elements 1 "all"
```

```
*duplicatemark elements 1 1
*createmark elements 1  $countnum(6)-$countnum(7)
*createvector 1 0.0000 0.0000 1.0000
*translatemark elements 1 1 -$blcl
*createmark elements 1  $countnum(6)-$countnum(7)
*createplane 1 0.0000 0.0000 1.0000 0.0000 0.0000 -$blcc
*reflectmark elements 1 1

#organize elements

set countnum(8) [expr $countnum(2)+95]
set countnum(9) [expr $countnum(4)+95]
set countnum(10) [expr $countnum(9)+257]
set countnum(11) [expr $countnum(10)+95]
set countnum(12) [expr $countnum(6)+95]
set countnum(13) [expr $countnum(12)+257]
set countnum(14) [expr $countnum(13)+95]
set countnum(15) [expr $countnum(14)+257]
set countnum(16) [expr $countnum(15)+95]
set countnum(17) [expr $countnum(16)+257]
set countnum(18) [expr $countnum(17)+95]


*createmark elements 1 $countnum(2)-$countnum(8) $countnum(4)-$countnum(9)
    $countnum(10)-$countnum(11) $countnum(6)-$countnum(12) $countnum(13)-
    $countnum(14) $countnum(15)-$countnum(16) $countnum(17)-$countnum(18)
*movemark elements 1 "matrix"

#merge edge nodes
*createmark elements 1  "all"
*equivalence elements 1 1e-006 1 0 0

#renumber nodes and elements

*createmark nodes 1 "all"
```

```
*renumber nodes 1 1 1 0 0
*createmark elements 1  "all"
*renumber elements 1 1 1 0 0


#display control
*window 0 0 0 0 0
*displaycollectorwithfilter components "off" "matrix" 0 1
*displaycollectorwithfilter components "off" "particle" 0 1
*view "iso1"
*setdisplayattributes 2 0

#create material & properties (follow Ansys template)

*collectorcreate materials "MAT1" "" 4
*createmark materials 1  "MAT1"
*renumber materials 1 1 1 0 0
*createmark materials 1  "MAT1"
*dictionaryload materials 1 "C:/Program Files/Altair/11.0/templates/feoutput
    /ansys/ansys.tpl" "MATERIAL"
*attributeupdateint materials 1 504 8 0 0 1
*createdoublearray 1  $TREF(present)
*attributeupdatedoublearray materials 1 505 8 2 0 1 1
*attributeupdateint materials 1 502 8 2 0 1
*attributeupdateint materials 1 56 8 0 0 1
*attributeupdateint materials 1 2644 8 0 0 1
*createdoublearray 1  $EX1(present)
*attributeupdatedoublearray materials 1 2645 8 2 0 1 1
*attributeupdateint materials 1 76 8 0 0 1
*attributeupdateint materials 1 2571 8 0 0 1
*createdoublearray 1  $PRXY1(present)
*attributeupdatedoublearray materials 1 2611 8 2 0 1 1
*attributeupdateint materials 1 502 8 2 0 1
*attributeupdateint materials 1 53 8 0 0 1
*attributeupdateint materials 1 2579 8 0 0 1
```

```
*createdoublearray 1 $DENS1(present)
*attributeupdatedoublearray materials 1 2619 8 2 0 1 1

*collectorcreate materials "MAT2" "" 3
*createmark materials 1  "MAT2"
*renumber materials 1 2 1 0 0
*createmark materials 1  "MAT2"
*dictionaryload materials 1 "C:/Program Files/Altair/11.0/templates/feoutput
    /ansys/ansys.tpl" "MATERIAL"
*attributeupdateint materials 2 504 8 0 0 1
*createdoublearray 1  $TREF(present)
*attributeupdatedoublearray materials 2 505 8 2 0 1 1
*attributeupdateint materials 2 502 8 2 0 1
*attributeupdateint materials 2 56 8 0 0 1
*attributeupdateint materials 2 2644 8 0 0 1
*createdoublearray 1  $EX2(present)
*attributeupdatedoublearray materials 2 2645 8 2 0 1 1
*attributeupdateint materials 2 76 8 0 0 1
*attributeupdateint materials 2 2571 8 0 0 1
*createdoublearray 1  $PRXY2(present)
*attributeupdatedoublearray materials 2 2611 8 2 0 1 1
*attributeupdateint materials 2 502 8 2 0 1
*attributeupdateint materials 2 53 8 0 0 1
*attributeupdateint materials 2 2579 8 0 0 1
*createdoublearray 1 $DENS2(present)
*attributeupdatedoublearray materials 2 2619 8 2 0 1 1

*collectorcreateonly properties "PROP1" "" 5
*collectorcreateonly properties "PROP2" "" 6

#assign materials & properties to components

*createmark components 1  "particle"
*materialupdate components 1 "MAT1"
*clearmark components 1
```

```
*createmark components 1  "particle"
*propertyupdate components 1 "PROP1"
*clearmark components 1
*createmark components 1  "matrix"
*materialupdate components 1 "MAT2"
*clearmark components 1
*createmark components 1  "matrix"
*propertyupdate components 1 "PROP2"
*clearmark components 1
}

::hwt::UnpostWindow ucparam;
};
```

A.8 The Source Code of *customUC* Tcl File

```
#####################################################################
## Filename: customUC.tcl
## Purpose:  Build up micromechanics models
## Function: VAMUCH & HyperWorks Interface
## Copyright (C) 2013 by Wenbin Yu, and Chong Teng.
## Support: Chong Teng <chongteng@aggiemail.usu.edu>
##
#####################################################################



namespace eval ::altair::customUC \
{
    variable recess;
    variable fillType;
    variable line_list_1;
    variable line_length;
    variable line_index;
```

```
    #variable line_index_2;
    variable i;
    variable node_list;
    #variable node_list_2;
    variable line_length;
    variable node_index_1;
    variable node_index_2;
    variable j;
    variable k;
    variable m;
    variable node_index_1_x;
    variable node_index_1_y;
    variable node_index_1_z;
    variable node_index_2_x;
    variable node_index_2_y;
    variable node_index_2_z;
    variable coordflag;
    variable edgeflag;

}


namespace eval ::altair::autoparam\
{
    variable recess;

    array set X_dir \
    {
        present          0
        past             0
    }
    array set Y_dir \
    {
        present          0
        past             0
```

```
    }
    array set Z_dir \
    {
        present          0
        past             0
    }
    array set X_base \
    {
        present          0
        past             0
    }
    array set Y_base \
    {
        present          0
        past             0
    }
    array set Z_base \
    {
        present          0
        past             0
    }

}

proc ::altair::customUC::customUCdialog {} {

    if {[winfo exists .custUCopt]} {
        destroy .custUCopt;
    }

    variable recess;

    set miny 150
    if {![::hwt::OnPc]} {set miny 175};
#####################################################################
```

```
## create window and buttons
#################################################################
    ::hwt::CreateWindow custUCopt \
        -windowtitle   "Customize Unit Cell Options" \
        -cancelButton   "Close" \
        -cancelFunc     ::altair::customUC::Quit \
        -addButton  Apply ::altair::customUC::Select no_icon \
        -minsize 350 $miny  \
        post;
    set xloc [ ::fepre::GetXLocation 350 ];
    set yloc [ ::fepre::GetYLocation 190 ];
    if {$yloc < 0} {set yloc 100};   ##yloc coming out negative on linux
        sometimes
    wm geometry .custUCopt +$xloc+$yloc;

    KeepOnTop   .custUCopt

    set recess [ ::hwt::WindowRecess custUCopt ];

    grid columnconfigure $recess 1 -weight 3

set ::altair::customUC::fillType manual;

    radiobutton $recess.manual \
        -text "Manual" \
        -variable ::altair::customUC::fillType \
        -value manual \
        -state normal;

    radiobutton $recess.automatic \
        -text "Automatic" \
        -variable ::altair::customUC::fillType \
        -value automatic \
        -state normal;
```

```
    label $recess.l1\
       -text "Choose the Way to Apply Periodic Boundary Conditions :" \
       -state normal;

   grid $recess.l1     -row 1 -column 0 -padx 20 -pady 15 -sticky nw;
    grid $recess.manual  -row 2 -column 0 -padx 20 -pady 10 -sticky nw;
    grid $recess.automatic  -row 3 -column 0 -padx 20 -pady 10 -sticky nw;


::hwt::RemoveDefaultButtonBinding $recess

};

#Call the function
::altair::customUC::customUCdialog

proc ::altair::customUC::Quit {} \
{
    ::hwt::UnpostWindow custUCopt
    hm_usermessage "";
};

proc ::altair::customUC::Select {} \
{

    variable fillType;
    variable namemark;
    variable recess;



if { $fillType == "manual" } \
{
 ::altair::customUC::Quit  ;
 set miny 150
```

```
    if {![::hwt::OnPc]} {set miny 175};
#################################################################
## create window and buttons
#################################################################
    ::hwt::CreateWindow manuparamopt \
        -windowtitle   "Manually Apply Periodic Boundary Condition" \
        -cancelButton   "Return" \
        -cancelFunc     ::altair::customUC::Quit_2 \
        -addButton  Apply ::altair::customUC::Select_2 no_icon \
        -minsize 350 $miny  \
        post;
    set xloc [ ::fepre::GetXLocation 350 ];
    set yloc [ ::fepre::GetYLocation 190 ];
    if {$yloc < 0} {set yloc 100};   ##yloc coming out negative on linux
        sometimes
    wm geometry .manuparamopt +$xloc+$yloc;

    KeepOnTop   .manuparamopt

    set recess [ ::hwt::WindowRecess manuparamopt ];

    grid columnconfigure $recess 1 -weight 3

    label $recess.l1\
       -text "Select Periodic Edges:" \
       -state normal;
    label $recess.l2 \
       -text "Notice:"\
       -bg yellow\
       -state normal;
    label $recess.l3 \
       -text "If your model has more than one lines on a single edge"\
       -bg yellow\
       -state normal;
     label $recess.l4 \
```

```
        -text "you will need to add rigidlink between nodes manually. " \
        -bg yellow\
        -state normal;


    button $recess.b1 \
        -text "Select One Set of Periodic Edges" \
        -height [::hwt::DluHeight 1] \
        -width [::hwt::DluHeight 20] \
        -command ::altair::customUC::SelectLines \
        -bg yellow \
        -state normal;

   grid $recess.l1     -row 1 -column 0 -padx 10 -pady 15 -sticky nw;
   grid $recess.b1     -row 2 -column 0 -padx 60 -pady 15 -sticky nw;
   grid $recess.l2     -row 3 -column 0 -padx 10 -pady 15 -sticky nw;
   grid $recess.l3     -row 4 -column 0 -padx 40 -pady 0 -sticky nw;
   grid $recess.l4     -row 5 -column 0 -padx 40 -pady 0 -sticky nw;
} \
elseif { $fillType == "automatic" } \
{
::altair::customUC::Quit   ;
    if {[winfo exists .autoparamopt ]} {
        destroy .autoparamopt ;
    }

    set miny 150
    if {![::hwt::OnPc]} {set miny 175};
#####################################################################
## create window and buttons
#####################################################################
    ::hwt::CreateWindow autoparamopt \
        -windowtitle   "Automatically Apply Periodic Boundary Condition" \
        -cancelButton   "Close" \
        -cancelFunc     ::altair::autoparam::Quit \
```

```
        -addButton  Apply ::altair::autoparam::Select no_icon \
        -minsize 350 $miny  \
        post;
    set xloc [ ::fepre::GetXLocation 350 ];
    set yloc [ ::fepre::GetYLocation 190 ];
    if {$yloc < 0} {set yloc 100};   ##yloc coming out negative on linux
       sometimes
    wm geometry .autoparamopt +$xloc+$yloc;

    KeepOnTop   .autoparamopt

    set recess [ ::hwt::WindowRecess autoparamopt ];

    grid columnconfigure $recess 1 -weight 3

label $recess.l1 -text "Input Periodic Boundary Condition Information:" -
   state normal;
label $recess.l2 -text "Dimension along X Direction:" -state normal;
label $recess.l3 -text "Dimension along Y Direction:" -state normal;
label $recess.l4 -text "Dimension along Z Direction:" -state normal;
label $recess.l5 -text "Origin of the Model:" -state normal;
label $recess.l6 -text "Origin X Coordinate (X0):" -state normal;
label $recess.l7 -text "Origin Y Coordinate (Y0):" -state normal;
label $recess.l8 -text "Origin Z Coordinate (Z0):" -state normal;
label $recess.l9 \
       -text "Use Zero if Certain Dimension is Not Exist in Model" -bg
          yellow -state normal;
entry $recess.e1 \
            -textvariable ::altair::autoparam::X_dir(present) \
            -width 10 -justify left \
            -state normal;
        set ::altair::autoparam::X_dir(present) 0;
entry $recess.e2 \
            -textvariable ::altair::autoparam::Y_dir(present) \
            -width 10 -justify left \
```

```
            -state normal;
        set ::altair::autoparam::Y_dir(present) 0;
entry $recess.e3 \
            -textvariable ::altair::autoparam::Z_dir(present) \
            -width 10 -justify left \
            -state normal;
        set ::altair::autoparam::Z_dir(present) 0;
entry $recess.e4 \
            -textvariable ::altair::autoparam::X_base(present) \
            -width 10 -justify left \
            -state normal;
        set ::altair::autoparam::X_base(present) 0;
entry $recess.e5 \
            -textvariable ::altair::autoparam::Y_base(present) \
            -width 10 -justify left \
            -state normal;
        set ::altair::autoparam::Y_base(present) 0;
entry $recess.e6 \
            -textvariable ::altair::autoparam::Z_base(present) \
            -width 10 -justify left \
            -state normal;
        set ::altair::autoparam::Z_base(present) 0;


grid $recess.l1     -row 1 -column 0 -padx 20 -pady 8 -sticky nw;
grid $recess.l2     -row 2 -column 0 -padx 40 -pady 8 -sticky nw;
grid $recess.e1     -row 2 -column 1 -pady 8 -sticky nw;
grid $recess.l3     -row 3 -column 0 -padx 40 -pady 8 -sticky nw;
grid $recess.e2     -row 3 -column 1 -pady 8 -sticky nw;
grid $recess.l4     -row 4 -column 0 -padx 40 -pady 8 -sticky nw;
grid $recess.e3     -row 4 -column 1 -pady 8 -sticky nw;
grid $recess.l9     -row 5 -column 0 -padx 20 -pady 4 -sticky nw;
grid $recess.l5     -row 6 -column 0 -padx 20 -pady 15 -sticky nw;
grid $recess.l6     -row 7 -column 0 -padx 40 -pady 8 -sticky nw;
grid $recess.e4     -row 7 -column 1 -pady 8 -sticky nw;
```

```
grid $recess.l7     -row 8 -column 0 -padx 40 -pady 8 -sticky nw;
grid $recess.e5     -row 8 -column 1 -pady 8 -sticky nw;
grid $recess.l8     -row 9 -column 0 -padx 40 -pady 8 -sticky nw;
grid $recess.e6     -row 9 -column 1 -pady 8 -sticky nw;

}
};

proc ::altair::customUC::SelectLines {} \
{
     variable recess;
     variable line_list_1;

::hwt::UnpostWindow manuparamopt;
*clearmark lines 1
*createlistpanel lines 1 "Select One Set of Periodic Edges"
set line_list_1 [hm_getlist lines 1];

if {![Null line_list_1]} {
$recess.b1 config -bg green;
} else {
$recess.b1 config -bg yellow;
}
::hwt::PostWindow manuparamopt;
};


proc ::altair::customUC::Quit_2 {} \
{
    ::hwt::UnpostWindow manuparamopt;
    ::altair::customUC::customUCdialog;
    hm_usermessage "";
};
```

```
proc ::altair::customUC::Select_2 {}\
{
    variable line_list_1;
    variable line_length;
    variable line_index;
    #variable line_index_2;
    variable i;
    variable node_list;
    #variable node_list_2;
    variable line_length;
    variable node_index_1;
    variable node_index_2;
    variable j;
    variable k;
    variable m;
    variable node_index_1_x;
    variable node_index_1_y;
    variable node_index_1_z;
    variable node_index_2_x;
    variable node_index_2_y;
    variable node_index_2_z;
    variable coordflag;
    variable edgeflag;

set line_length [llength [hm_getlist lines 1]]

for {set i 1} {$i<=$line_length} {incr i} {
    set line_index($i) [lindex $line_list_1 [expr $i-1]]
    hm_createmark nodes 1 "by lines" $line_index($i);
    set node_list($i) [hm_getmarkvalue nodes 1 id 0];
    set node_length [llength [hm_getmarkvalue nodes 1 id 0]];

for {set k 1} {$k<=$i-1} {incr k} {
    for {set j 1} {$j<=$node_length} {incr j} {
    set node_index_1 [lindex $node_list($i) [expr $j-1]]
```

```
    set node_index_1_x [hm_getentityvalue nodes $node_index_1 "x" 0]
    set node_index_1_y [hm_getentityvalue nodes $node_index_1 "y" 0]
    set node_index_1_z [hm_getentityvalue nodes $node_index_1 "z" 0]
    set edgeflag 0;

    for {set m 1} {$m<=$node_length} {incr m} {
    set node_index_2 [lindex $node_list($k) [expr $m-1]]

    set coordflag 0;

    set node_index_2_x [hm_getentityvalue nodes $node_index_2 "x" 0]
    set node_index_2_y [hm_getentityvalue nodes $node_index_2 "y" 0]
    set node_index_2_z [hm_getentityvalue nodes $node_index_2 "z" 0]

if {[expr abs($node_index_1_x-$node_index_2_x)<=0.0001]} {
set coordflag [expr $coordflag+1]
}
if {[expr abs($node_index_1_y-$node_index_2_y)<=0.0001]}  {
set coordflag [expr $coordflag+1]
}
if {[expr abs($node_index_1_z-$node_index_2_z)<=0.0001]}  {
set coordflag [expr $coordflag+1]
}
if {$coordflag<=1} {
set edgeflag [expr $edgeflag+1]
}
if {$coordflag==2} {
*rigidlinkwithset_twonodes $node_index_1 $node_index_2 123456;
}
if {$edgeflag==$node_length} {
break
}
}
}
}
```

```
}
*createmark elements 1 "by config" "rigidlink"
*maskentitymark elements 1 0
hm_usermessage "The Periodic Boundary Condition has been Successfully Added
    ";

::hwt::UnpostWindow manuparamopt;

};

proc ::altair::autoparam::Quit {} \
{
    ::hwt::UnpostWindow autoparamopt;
    ::altair::customUC::customUCdialog;
    hm_usermessage "";
};

proc ::altair::autoparam::Select {} \
{
variable X_dir;
variable Y_dir;
variable Z_dir;
variable X_base;
variable Y_base;
variable Z_base;
variable countnum;
variable node_list_1;
variable node_list_2;
variable node_length;
variable node_index_1;
variable node_index_2;
variable i;
variable node_index_1_x;
variable node_index_1_y;
variable node_index_1_z;
```

```
variable node_index_2_x;
variable node_index_2_y;
variable node_index_2_z;
variable coordflag;
variable j;

set countnum(1) [expr $X_base(present)+$X_dir(present)/2.0000]
set countnum(2) [expr $X_base(present)-$X_dir(present)/2.0000]
set countnum(3) [expr $Y_base(present)+$Y_dir(present)/2.0000]
set countnum(4) [expr $Y_base(present)-$Y_dir(present)/2.0000]
set countnum(5) [expr $Z_base(present)+$Z_dir(present)/2.0000]
set countnum(6) [expr $Z_base(present)-$Z_dir(present)/2.0000]

# couple nodes by adding 1D rigid bar elements between them

if {$X_dir(present)!=0} {
*createmark nodes 1 "on plane" $countnum(1) $Y_base(present) $Z_base(present
    ) 1 0 0 .0004 1 1
set node_list_1 [hm_getmark nodes 1]
set node_length [llength [hm_getmark nodes 1]]
*createmark nodes 2 "on plane" $countnum(2) $Y_base(present) $Z_base(present
    ) 1 0 0 .0004 1 1
set node_list_2 [hm_getmark nodes 2]

for {set i 1} {$i<=$node_length} {incr i} {

    set node_index_1 [lindex $node_list_1 [expr $i-1]]

    set node_index_1_y [hm_getentityvalue nodes $node_index_1 "y" 0]
    set node_index_1_z [hm_getentityvalue nodes $node_index_1 "z" 0]

for {set m 1} {$m<=$node_length} {incr m} {

    set node_index_2 [lindex $node_list_2 [expr $m-1]]
```

```
    set coordflag 0;

    set node_index_2_y [hm_getentityvalue nodes $node_index_2 "y" 0]
    set node_index_2_z [hm_getentityvalue nodes $node_index_2 "z" 0]

if {[expr abs($node_index_1_y-$node_index_2_y)<=0.0001]}  {
set coordflag [expr $coordflag+1]
}
if {[expr abs($node_index_1_z-$node_index_2_z)<=0.0001]}  {
set coordflag [expr $coordflag+1]
}
if {$coordflag==2} {
*rigidlinkwithset_twonodes $node_index_1 $node_index_2 123456;
}

}
}



}
if {$Y_dir(present)!=0} {
*createmark nodes 1 "on plane" $X_base(present) $countnum(3) $Z_base(present
    ) 0 1 0 .0004 1 1
set node_list_1 [hm_getmark nodes 1]
set node_length [llength [hm_getmark nodes 1]]
*createmark nodes 2 "on plane" $X_base(present) $countnum(4) $Z_base(present
    ) 0 1 0 .0004 1 1
set node_list_2 [hm_getmark nodes 2]

for {set i 1} {$i<=$node_length} {incr i} {

    set node_index_1 [lindex $node_list_1 [expr $i-1]]

    set node_index_1_x [hm_getentityvalue nodes $node_index_1 "x" 0]
    set node_index_1_z [hm_getentityvalue nodes $node_index_1 "z" 0]
```

```
for {set m 1} {$m<=$node_length} {incr m} {

    set node_index_2 [lindex $node_list_2 [expr $m-1]]

    set coordflag 0;

    set node_index_2_x [hm_getentityvalue nodes $node_index_2 "x" 0]
    set node_index_2_z [hm_getentityvalue nodes $node_index_2 "z" 0]

if {[expr abs($node_index_1_x-$node_index_2_x)<=0.0001]}  {
set coordflag [expr $coordflag+1]
}
if {[expr abs($node_index_1_z-$node_index_2_z)<=0.0001]}  {
set coordflag [expr $coordflag+1]
}
if {$coordflag==2} {
*rigidlinkwithset_twonodes $node_index_1 $node_index_2 123456;
}

}
}

}

if {$Z_dir(present)!=0} {
*createmark nodes 1 "on plane" $X_base(present) $Y_base(present) $countnum
    (5) 0 0 1 .0004 1 1
set node_list_1 [hm_getmark nodes 1]
set node_length [llength [hm_getmark nodes 1]]
*createmark nodes 2 "on plane" $X_base(present) $Y_base(present) $countnum
    (6) 0 0 1 .0004 1 1
set node_list_2 [hm_getmark nodes 2]
```

```
for {set i 1} {$i<=$node_length} {incr i} {

    set node_index_1 [lindex $node_list_1 [expr $i-1]]

    set node_index_1_x [hm_getentityvalue nodes $node_index_1 "x" 0]
    set node_index_1_y [hm_getentityvalue nodes $node_index_1 "y" 0]

for {set m 1} {$m<=$node_length} {incr m} {

    set node_index_2 [lindex $node_list_2 [expr $m-1]]

    set coordflag 0;

    set node_index_2_x [hm_getentityvalue nodes $node_index_2 "x" 0]
    set node_index_2_y [hm_getentityvalue nodes $node_index_2 "y" 0]

if {[expr abs($node_index_1_x-$node_index_2_x)<=0.0001]}  {
set coordflag [expr $coordflag+1]
}
if {[expr abs($node_index_1_y-$node_index_2_y)<=0.0001]}  {
set coordflag [expr $coordflag+1]
}
if {$coordflag==2} {
*rigidlinkwithset_twonodes $node_index_1 $node_index_2 123456;
}

}
}

}

*createmark elements 1 "by config" "rigidlink"
*maskentitymark elements 1 0
hm_usermessage "The Periodic Boundary Condition has been Successfully Added
    ";
```

```
::hwt::UnpostWindow autoparamopt
};
```

A.9 The Source Code of *microtype* Tcl File

```
#####################################################################
## Filename: microtype.tcl
## Purpose:  Define the type of micromechanics analysis and start ##
    generating input file for VAMUCH
## Function: VAMUCH & HyperWorks Interface
## Copyright (C) 2013 by Wenbin Yu, and Chong Teng.
## Support: Chong Teng <chongteng@aggiemail.usu.edu>
##
#####################################################################


namespace eval ::altair::microtype \
{
    variable recess;
    variable microType;
    variable constType;
    variable tempFlag;
    variable flag;
    variable altair_dir;
    variable usr_dir;
    array set VolTot \
    {
        present          0
        past             0
    }
array set Disp_m1 \
    {
        present          0
        past             0
```

```
}
array set Disp_m2 \
{
    present          0
    past             0
}
array set Disp_m3 \
{
    present          0
    past             0
}
array set DisGrad_11 \
{
    present          0
    past             0
}
array set DisGrad_12 \
{
    present          0
    past             0
}
array set DisGrad_13 \
{
    present          0
    past             0
}
array set DisGrad_21 \
{
    present          0
    past             0
}
array set DisGrad_22 \
{
    present          0
    past             0
```

```
}
array set DisGrad_23 \
{
    present          0
    past             0
}
array set DisGrad_31 \
{
    present          0
    past             0
}
array set DisGrad_32 \
{
    present          0
    past             0
}
array set DisGrad_33 \
{
    present          0
    past             0
}

array set Temp_m \
{
    present          0
    past             0
}
array set Temp_1 \
{
    present          0
    past             0
}
array set Temp_2 \
{
    present          0
```

```
        past             0
    }
    array set Temp_3 \
    {
        present          0
        past             0
    }
}

namespace eval ::altair::analyparam \
{
    variable recess;
}

namespace eval ::altair::recovparam \
{
    variable recess;
}


proc ::altair::microtype::MicrotypeDialog {} {

    if {[winfo exists .microanalyopt]} {
        destroy .microanalyopt;
    }

    variable    recess;

#####################################################################
## create window and buttons
#####################################################################
    ::hwt::CreateWindow microanalyopt \
        -windowtitle   "Micromechanics Analysis Type" \
        -cancelButton   "Close" \
        -cancelFunc     ::altair::microtype::Quit \
```

```
        -addButton  OK ::altair::microtype::OK no_icon \
        -minsize 350 150  \
        post;
    set xloc [ ::fepre::GetXLocation 350 ];
    set yloc [ ::fepre::GetYLocation 190 ];
    if {$yloc < 0} {set yloc 100};   ##yloc coming out negative on linux
       sometimes
    wm geometry .microanalyopt +$xloc+$yloc;

    KeepOnTop   .microanalyopt

    set recess [ ::hwt::WindowRecess microanalyopt ];

    grid columnconfigure $recess 1 -weight 3

    set ::altair::microtype::microType 0;

#####################################################################
## define option layouts
#####################################################################

    radiobutton $recess.effanalysis \
        -text "Effective Properties Analysis" \
        -variable ::altair::microtype::microType \
        -value 0 \
        -state normal;

    radiobutton $recess.rcvanalysis \
        -text "Recovery Analysis" \
        -variable ::altair::microtype::microType \
        -value 1 \
        -state normal;

    label $recess.l1 \
       -text "Choose Micro-Analysis Type:" -state normal;
```

```
    label $recess.l2 \
       -text "Notice:" -bg yellow -state normal;

    label $recess.l3 \
       -text "Effective Properties Analysis Must be Done before Recovery
           Analysis" -bg yellow -state normal;

#####################################################################
## position of the options
#####################################################################
   grid $recess.l1     -row 1 -column 0 -padx 5 -pady 5 -sticky nw;
    grid $recess.effanalysis -row 2 -column 0 -padx 20 -pady 10 -sticky nw;
    grid $recess.rcvanalysis -row 3 -column 0 -padx 20 -pady 10 -sticky nw;
    grid $recess.l2     -row 4 -column 0 -padx 0 -pady 5 -sticky nw;
    grid $recess.l3     -row 5 -column 0 -padx 0 -pady 5 -sticky nw;

    ::hwt::RemoveDefaultButtonBinding $recess
};


proc ::altair::microtype::Quit {} \
{
    ::hwt::UnpostWindow microanalyopt
    hm_usermessage "";
};

#Call the function
::altair::microtype::MicrotypeDialog

proc ::altair::microtype::OK {} \
{
::hwt::UnpostWindow microanalyopt
::altair::microtype::Inputgenerate
};
```

```
proc ::altair::microtype::Inputgenerate {}\
{
    variable microType;

    ::altair::microtype::Quit;
    ::hwt::CreateWindow analyparam \
       -windowtitle  "Constitutive Analysis Type"\
       -cancelButton   "Close" \
       -cancelFunc     ::altair::analyparam::Quit \
       -addButton OK ::altair::microtype::FinalOK no_icon \
       -addButton Return ::altair::analyparam::Return no_icon \
       -minsize 350 150  \
        post;
    set xloc [ ::fepre::GetXLocation 350 ];
    set yloc [ ::fepre::GetYLocation 190 ];
    if {$yloc < 0} {set yloc 100};   ##yloc coming out negative on linux
        sometimes
    wm geometry .analyparam +$xloc+$yloc;

    KeepOnTop   .analyparam

    set recess [ ::hwt::WindowRecess analyparam ];

    grid columnconfigure $recess 1 -weight 3

    set ::altair::microtype::constType 0;
    set ::altair::microtype::tempFlag 0;

    label $recess.l4 -text "Choose the Type of Constitutive Analysis:" -
        state normal;
    label $recess.l5 -text "Choose whether the Model Temperature
        Distribution is Uniform:" -state normal;
    label $recess.l6 -text "Enter the Volume of the Whole Structure:" -state
         normal;
```

```
 label $recess.l7 \
   -text "Notice: Do Not Subtract Void Volume" -bg yellow -state normal;
entry $recess.e1 \
        -textvariable ::altair::microtype::VolTot(present)\
        -width 10 -justify left \
        -state normal;
    set ::altair::microtype::VolTot(present) 1.0;

radiobutton $recess.elastanalysis \
    -text "Elastic Analysis" \
    -variable ::altair::microtype::constType \
    -value 0 \
    -state normal;

radiobutton $recess.thermoanalysis \
    -text "Thermoelastic Analysis" \
    -variable ::altair::microtype::constType \
    -value 1 \
    -state normal;

radiobutton $recess.conductanalysis \
    -text "Conduction Analysis" \
    -variable ::altair::microtype::constType \
    -value 2 \
    -state normal;

radiobutton $recess.tempindepend \
    -text "Uniform Temperature Distribution" \
    -variable ::altair::microtype::tempFlag \
    -value 0 \
    -state normal;

radiobutton $recess.tempdepend \
    -text "Non-uniform Temperature Distribution" \
    -variable ::altair::microtype::tempFlag \
```

```
        -value 1 \
        -state normal;


    grid $recess.l4     -row 1 -column 0 -padx 5 -pady 5 -sticky nw;
    grid $recess.elastanalysis -row 2 -column 0 -padx 20 -pady 5 -sticky nw;
    grid $recess.thermoanalysis -row 3 -column 0 -padx 20 -pady 5 -sticky nw
        ;
    grid $recess.conductanalysis -row 4 -column 0 -padx 20 -pady 5 -sticky
        nw;
    grid $recess.l5     -row 5 -column 0 -padx 5 -pady 5 -sticky nw;
    grid $recess.tempindepend -row 6 -column 0 -padx 20 -pady 10 -sticky nw;
    grid $recess.tempdepend -row 7 -column 0 -padx 20 -pady 10 -sticky nw;
    grid $recess.l6     -row 8 -column 0 -padx 5 -pady 5 -sticky nw;
    grid $recess.e1     -row 9 -column 0 -padx 200 -pady 5 -sticky nw;
    grid $recess.l7     -row 10 -column 0 -padx 5 -pady 5 -sticky nw;
};

proc ::altair::analyparam::Quit {} \
{
    ::hwt::UnpostWindow analyparam;
};

proc ::altair::analyparam::Return {} \
{
    ::hwt::UnpostWindow analyparam;
    ::hwt::PostWindow microanalyopt;
};

proc ::altair::microtype::FinalOK {} \
{
variable microType;
variable constType;
variable tempFlag;
variable npairFlag;
```

```
variable totalelems;
variable nelems;
variable VolTot;
variable altair_dir;
variable usr_dir;
variable rootname;

 if { $microType == "0" }\
 {

set npairFlag [hm_count elems all 55 0];
set totalelems [hm_count elems all 0 0];
set nelems [expr $totalelems-$npairFlag]
*createmark undef 1;
*createdoublearray 6 $microType $constType $tempFlag $npairFlag $nelems
    $VolTot(present);
*metadatamarkremove undef 1 "flag"
*metadatamarkdoublearray undef 1 "flag" 1 6;

set altair_dir [hm_info -appinfo ALTAIR_HOME];
set usr_dir [hm_info -appinfo CURRENTWORKINGDIR];
set rootname [file rootname [file tail [hm_info currentfile]]];
if {[llength $rootname] == "0" }\
{set rootname Untitled}

*feoutputwithdata "$altair_dir/templates/feoutput/vamuch/const_analy.tpl" "
    $usr_dir/$rootname.vam" 0 0 2 1 0

}
 if { $microType == "1" }\
 {
    ::altair::analyparam::Quit;
    ::hwt::CreateWindow recovparam \
       -windowtitle  "Input Macro Fields"\
       -cancelButton   "Close" \
```

```
   -cancelFunc     ::altair::recovparam::Quit \
   -addButton OK ::altair::microtype::RecovOK no_icon \
   -addButton Return ::altair::recovparam::Return no_icon \
   -minsize 600 100  \
    post;
set xloc [ ::fepre::GetXLocation 600 ];
set yloc [ ::fepre::GetYLocation 100 ];
if {$yloc < 0} {set yloc 100};   ##yloc coming out negative on linux
    sometimes
wm geometry .recovparam +$xloc+$yloc;

KeepOnTop   .recovparam

set recess [ ::hwt::WindowRecess recovparam ];

grid columnconfigure $recess 1 -weight 3

label $recess.l1 -text "V_1,V_2,V_3 (Displacment):" -state normal;
label $recess.l2 -text "V_11,V_12,V_13 (Disp Gradient):"  -state normal;
label $recess.l3 -text "V_21,V_22,V_23 (Disp Gradient):"  -state normal;
label $recess.l4 -text "V_31,V_32,V_33 (Disp Gradient):"  -state normal;
label $recess.l5 -text "T_m (Temperature):"  -state normal;
label $recess.l6 -text "T_1,T_2,T_3 (Temp Gradient):"  -state normal;
entry $recess.e1 \
        -textvariable ::altair::microtype::Disp_m1(present)\
        -width 8 -justify left \
        -state normal;
    set ::altair::microtype::Disp_m1(present) 0.0;
entry $recess.e2 \
        -textvariable ::altair::microtype::Disp_m2(present)\
        -width 8 -justify left \
        -state normal;
    set ::altair::microtype::Disp_m2(present) 0.0;
entry $recess.e3 \
        -textvariable ::altair::microtype::Disp_m3(present)\
```

```
        -width 8 -justify left \
        -state normal;
    set ::altair::microtype::Disp_m3(present) 0.0;
entry $recess.e4 \
        -textvariable ::altair::microtype::DisGrad_11(present)\
        -width 8 -justify left \
        -state normal;
    set ::altair::microtype::DisGrad_11(present) 0.0;
entry $recess.e5 \
        -textvariable ::altair::microtype::DisGrad_12(present)\
        -width 8 -justify left \
        -state normal;
    set ::altair::microtype::DisGrad_12(present) 0.0;
entry $recess.e6 \
        -textvariable ::altair::microtype::DisGrad_13(present)\
        -width 8 -justify left \
        -state normal;
    set ::altair::microtype::DisGrad_13(present) 0.003;
entry $recess.e7 \
        -textvariable ::altair::microtype::DisGrad_21(present)\
        -width 8 -justify left \
        -state normal;
    set ::altair::microtype::DisGrad_21(present) 0.0;
entry $recess.e8 \
        -textvariable ::altair::microtype::DisGrad_22(present)\
        -width 8 -justify left \
        -state normal;
    set ::altair::microtype::DisGrad_22(present) 0.0;
entry $recess.e9 \
        -textvariable ::altair::microtype::DisGrad_23(present)\
        -width 8 -justify left \
        -state normal;
    set ::altair::microtype::DisGrad_23(present) 0.006;
entry $recess.e10 \
        -textvariable ::altair::microtype::DisGrad_31(present)\
```

```
        -width 8 -justify left \
        -state normal;
    set ::altair::microtype::DisGrad_31(present) 0.0;
entry $recess.e11 \
        -textvariable ::altair::microtype::DisGrad_32(present)\
        -width 8 -justify left \
        -state normal;
    set ::altair::microtype::DisGrad_32(present) 0.0;
entry $recess.e12 \
        -textvariable ::altair::microtype::DisGrad_33(present)\
        -width 8 -justify left \
        -state normal;
    set ::altair::microtype::DisGrad_33(present) 0.009;
entry $recess.e13 \
        -textvariable ::altair::microtype::Temp_m(present)\
        -width 8 -justify left \
        -state normal;
    set ::altair::microtype::Temp_m(present) 100.0;
entry $recess.e14 \
        -textvariable ::altair::microtype::Temp_1(present)\
        -width 8 -justify left \
        -state normal;
    set ::altair::microtype::Temp_1(present) 0.1;
entry $recess.e15 \
        -textvariable ::altair::microtype::Temp_2(present)\
        -width 8 -justify left \
        -state normal;
    set ::altair::microtype::Temp_2(present) 0.2;
entry $recess.e16 \
        -textvariable ::altair::microtype::Temp_3(present)\
        -width 8 -justify left \
        -state normal;
    set ::altair::microtype::Temp_3(present) 0.3;
```

```
 if { $constType == "0" }\
 {
  label $recess.l7 -text "Input Macro Displacement and Displacement Gradient
      :"  -state normal;

    grid $recess.l7     -row 1 -column 0 -padx 0 -pady 5 -sticky nw;
    grid $recess.l1     -row 2 -column 0 -padx 0 -pady 5 -sticky nw;
    grid $recess.e1     -row 2 -column 1 -padx 28 -pady 5 -sticky nw;
    grid $recess.e2     -row 2 -column 2 -padx 28 -pady 5 -sticky nw;
    grid $recess.e3     -row 2 -column 3 -padx 28 -pady 5 -sticky nw;
    grid $recess.l2     -row 3 -column 0 -padx 0 -pady 5 -sticky nw;
    grid $recess.e4     -row 3 -column 1 -padx 28 -pady 5 -sticky nw;
    grid $recess.e5     -row 3 -column 2 -padx 28 -pady 5 -sticky nw;
    grid $recess.e6     -row 3 -column 3 -padx 28 -pady 5 -sticky nw;
    grid $recess.l3     -row 4 -column 0 -padx 0 -pady 5 -sticky nw;
    grid $recess.e7     -row 4 -column 1 -padx 28 -pady 5 -sticky nw;
    grid $recess.e8     -row 4 -column 2 -padx 28 -pady 5 -sticky nw;
    grid $recess.e9     -row 4 -column 3 -padx 28 -pady 5 -sticky nw;
    grid $recess.l4     -row 5 -column 0 -padx 0 -pady 5 -sticky nw;
    grid $recess.e10     -row 5 -column 1 -padx 28 -pady 5 -sticky nw;
    grid $recess.e11     -row 5 -column 2 -padx 28 -pady 5 -sticky nw;
    grid $recess.e12     -row 5 -column 3 -padx 28 -pady 5 -sticky nw;
}
 if { $constType == "1" }\
 {
label $recess.l7 -text "Input Macro Dispacement, Disp_Grad and Temperature:"
     -state normal;

    grid $recess.l7     -row 1 -column 0 -padx 0 -pady 5 -sticky nw;
    grid $recess.l1     -row 2 -column 0 -padx 0 -pady 5 -sticky nw;
    grid $recess.e1     -row 2 -column 1 -padx 28 -pady 5 -sticky nw;
    grid $recess.e2     -row 2 -column 2 -padx 28 -pady 5 -sticky nw;
    grid $recess.e3     -row 2 -column 3 -padx 28 -pady 5 -sticky nw;
    grid $recess.l2     -row 3 -column 0 -padx 0 -pady 5 -sticky nw;
```

```
    grid $recess.e4     -row 3 -column 1 -padx 28 -pady 5 -sticky nw;
    grid $recess.e5     -row 3 -column 2 -padx 28 -pady 5 -sticky nw;
    grid $recess.e6     -row 3 -column 3 -padx 28 -pady 5 -sticky nw;
    grid $recess.l3     -row 4 -column 0 -padx 0 -pady 5 -sticky nw;
    grid $recess.e7     -row 4 -column 1 -padx 28 -pady 5 -sticky nw;
    grid $recess.e8     -row 4 -column 2 -padx 28 -pady 5 -sticky nw;
    grid $recess.e9     -row 4 -column 3 -padx 28 -pady 5 -sticky nw;
    grid $recess.l4     -row 5 -column 0 -padx 0 -pady 5 -sticky nw;
    grid $recess.e10     -row 5 -column 1 -padx 28 -pady 5 -sticky nw;
    grid $recess.e11     -row 5 -column 2 -padx 28 -pady 5 -sticky nw;
    grid $recess.e12     -row 5 -column 3 -padx 28 -pady 5 -sticky nw;
    grid $recess.l5     -row 6 -column 0 -padx 0 -pady 5 -sticky nw;
    grid $recess.e13     -row 6 -column 1 -padx 28 -pady 5 -sticky nw;

}
 if { $constType == "2" }\
 {
  label $recess.l7 -text "Input Macro Temperature and Temperature Gradient:"
       -state normal;

    grid $recess.l7     -row 1 -column 0 -padx 0 -pady 5 -sticky nw;
    grid $recess.l5     -row 2 -column 0 -padx 0 -pady 5 -sticky nw;
    grid $recess.e13     -row 2 -column 1 -padx 28 -pady 5 -sticky nw;
    grid $recess.l6     -row 3 -column 0 -padx 0 -pady 5 -sticky nw;
    grid $recess.e14     -row 3 -column 1 -padx 28 -pady 5 -sticky nw;
    grid $recess.e15     -row 3 -column 2 -padx 28 -pady 5 -sticky nw;
    grid $recess.e16     -row 3 -column 3 -padx 28 -pady 5 -sticky nw;
}
};

::hwt::UnpostWindow analyparam;
};

proc ::altair::recovparam::Quit {} \
{
```

```
    ::hwt::UnpostWindow recovparam;
};

proc ::altair::recovparam::Return {} \
{
    ::hwt::UnpostWindow recovparam;
    ::hwt::PostWindow analyparam;
};

proc ::altair::microtype::RecovOK {} \
{

variable microType;
variable constType;
variable tempFlag;
variable npairFlag;
variable totalelems;
variable nelems;
variable VolTot;
variable Disp_m1;
variable Disp_m2;
variable Disp_m3;
variable DisGrad_11;
variable DisGrad_12;
variable DisGrad_13;
variable DisGrad_21;
variable DisGrad_22;
variable DisGrad_23;
variable DisGrad_31;
variable DisGrad_32;
variable DisGrad_33;
variable Temp_m;
variable Temp_1;
variable Temp_2;
variable Temp_3;
```

```
variable altair_dir;
variable usr_dir;
variable rootname;

set npairFlag [hm_count elems all 55 0];
set totalelems [hm_count elems all 0 0];
set nelems [expr $totalelems-$npairFlag]
*createmark undef 1;
*createdoublearray 22 $microType $constType $tempFlag $npairFlag $nelems
    $VolTot(present) $Disp_m1(present) $Disp_m2(present) $Disp_m3(present)
    $DisGrad_11(present) $DisGrad_12(present) $DisGrad_13(present)
    $DisGrad_21(present) $DisGrad_22(present) $DisGrad_23(present)
    $DisGrad_31(present) $DisGrad_32(present) $DisGrad_33(present) $Temp_m(
    present) $Temp_1(present) $Temp_2(present) $Temp_3(present);
*metadatamarkremove undef 1 "flag"
*metadatamarkdoublearray undef 1 "flag" 1 22;

set altair_dir [hm_info -appinfo ALTAIR_HOME];
set usr_dir [hm_info -appinfo CURRENTWORKINGDIR];

set rootname [file rootname [file tail [hm_info currentfile]]];

if {[llength $rootname] == "0" }\
{set rootname Untitled}

*feoutputwithdata "$altair_dir/templates/feoutput/vamuch/const_analy.tpl" "
    $usr_dir/$rootname.vam" 0 0 2 1 0

::hwt::UnpostWindow recovparam;

}
```

A.10 The Source Code of *microsolve* Tcl File

```
namespace eval ::altair::microsolve \
```

```
{
variable microType;
variable usr_dir;
variable res_file;
variable res_vamfile;
variable u_data;
variable es_data;
variable ese_data;
variable data_clmn;
variable one;
variable node_num;
variable npairFlag;
variable totalelems;
variable nelems;
variable u_x;
variable u_y;
variable u_z;
variable node_count;
variable lgth_count;
variable cc;
variable dd;
variable nn_counter;
variable e_comp;
variable s_comp;
variable countnum;
variable rootname;
}


proc ::altair::microsolve::solve {} {

variable microType;
variable usr_dir;
variable res_file;
variable res_vamfile;
```

```
variable u_data;
variable es_data;
variable ese_data;
variable data_clmn;
variable one;
variable node_num;
variable npairFlag;
variable totalelems;
variable nelems;
variable u_x;
variable u_y;
variable u_z;
variable node_count;
variable lgth_count;
variable cc;
variable dd;
variable nn_counter;
variable e_comp;
variable s_comp;
variable countnum;
variable rootname;



set microType [hm_metadata findbyname flag undef]
set microType [lindex $microType 0 end]
set microType [lindex $microType 0]

set rootname [file rootname [file tail [hm_info currentfile]]];

if {[llength $rootname] == "0" }\
{set rootname Untitled}

if {$microType==0.0} {

file delete $rootname.vam.k
```

```
*systemcommand "vamuch $rootname.vam"

*systemcommand "NOTEPAD $rootname.vam.k"


hm_usermessage "The SwiftComp Micromechanics Effective Properties Analysis
    is Carried Out";
}

if {$microType==1.0} {
*systemcommand "vamuch $rootname.vam"

hm_usermessage "The SwiftComp Micromechanics Recovery Analysis is Carried
    Out";

set usr_dir [hm_info -appinfo CURRENTWORKINGDIR];
*writeh3dtofile "$usr_dir/$rootname.h3d" 1


}

};


::altair::microsolve::solve;
```

A.11 The Source Code of *const_analy* Template File

```
////////////////////////////////////////////////////////////////
// Filename: const_analy.tpl
// Purpose:  Template file to generate VAMUCH input file
// Function: VAMUCH & HyperWorks Interface
// Copyright (C) 2013 by Wenbin Yu, and Chong Teng.
// Support: Chong Teng <chongteng@aggiemail.usu.edu>
```

```
//
////////////////////////////////////////////////////////////////////

*realprecision(15)
*setelementcolorbypropsmethod(1)
*setelementcolorbymatsmethod(1)
*compressreal(1)
*globalmenuminimumstringlength(8)
*text()
    *scalefieldwidth(string,0)
*output()

*include(attrib.lst)


*metadata()
*format()
*treataslocal(counter20)
*if([@getentityvalue(elems, 1, config)=104])
*counterset(counter20,2)
*endif()
*if([@getentityvalue(elems, 1, config)=108])
*counterset(counter20,2)
*endif()
*if([@getentityvalue(elems, 1, config)=103])
*counterset(counter20,2)
*endif()
*if([@getentityvalue(elems, 1, config)=106])
*counterset(counter20,2)
*endif()
*if([@getentityvalue(elems, 1, config)=204])
*counterset(counter20,3)
*endif()
*if([@getentityvalue(elems, 1, config)=210])
*counterset(counter20,3)
```

```
*endif()
*if([@getentityvalue(elems, 1, config)=208])
*counterset(counter20,3)
*endif()
*if([@getentityvalue(elems, 1, config)=220])
*counterset(counter20,3)
*endif()
*field(integer,counter20,10)
*string("")
*string("  ")
*treataslocal(counter15)
*pointerset(pointer1,data,0)
*counterset(counter15,pointer1.pointervalue)
*field(integer,counter15,10)
*string("  ")
*treataslocal(counter14)
*pointerset(pointer1,data,1)
*counterset(counter14,pointer1.pointervalue)
*field(integer,counter14,10)
*string("  ")
*treataslocal(counter13)
*pointerset(pointer1,data,2)
*counterset(counter13,pointer1.pointervalue)
*field(integer,counter13,10)
*string("  ")
*treataslocal(counter7)
*counterset(counter7,0)
*field(integer,counter7,10)

*treataslocal(counter12)
*pointerset(pointer1,data,3)
*counterset(counter12,pointer1.pointervalue)
*treataslocal(counter18)
*pointerset(pointer1,data,4)
*counterset(counter18,pointer1.pointervalue)
```

```
*pointerset(pointer2,data,5)
*end()
*output()

*text()
*treataslocal(counter19)
*counterset(counter19,[@entitymaxid(nodes)])
*string(" ")
*field(integer,counter19,10)
*string("  ")
*field(integer,counter18,10)
*string(" ")
*field(integer,counter12,10)
*string("  ")
*treataslocal(counter17)
*counterset(counter17,[@entitymaxid(materials)])
*field(integer,counter17,10)
*string("  ")
*treataslocal(counter10)
*counterset(counter10,[@getentityvalue(mats, 1, $MPTEMP_LEN)])
*field(integer,counter10,10)
*string("  ")
*treataslocal(counter9)
*counterset(counter9,0)
*field(integer,counter9,10)
*string("  ")
*treataslocal(counter8)
*counterset(counter8,0)
*field(integer,counter8,10)
*string(" ")
*end()
*output()


*nodes()
```

```
*before()
*string("  ")
*sortnodes(byid)
*end()
*format()
*field(integer,id,10)
*string("  ")
*if([counter20==1])
*field(real,x,15)
*string("  ")
*end()
*endif()
*if([counter20==2])
*field(real,y,15)
*string("  ")
*field(real,z,15)
*string("  ")
*end()
*endif()
*if([counter20==3])
*field(real,x,15)
*string("  ")
*field(real,y,15)
*string("  ")
*field(real,z,15)
*string("  ")
*end()
*endif()
*output()

*elements(104,0,"Quad4","")

*before()
*sortelements(byid)
*end()
```

```
*format()
*field(integer,id,10)
*string("")
*field(integer,propertyid,5)
*string("  ")
*field(integer,node1.id,10)
*string(" ")
*field(integer,node2.id,10)
*string(" ")
*field(integer,node3.id,10)
*string(" ")
*field(integer,node4.id,10)
*string(" ")
*field(integer,0,10)
*string(" ")
*field(integer,0,10)
*string(" ")
*field(integer,0,10)
*string(" ")
*field(integer,0,10)
*string(" ")
*field(integer,0,10)
*string(" ")
*field(integer,0,10)
*string(" ")
*end()
*output()

*elements(108,0,"Quad8","")
*before()
*sortelements(byid)
*end()
*format()
*field(integer,id,10)
*string("")
*field(integer,propertyid,5)
*string("  ")
```

```
*field(integer,node1.id,10)
*string(" ")
*field(integer,node2.id,10)
*string(" ")
*field(integer,node3.id,10)
*string(" ")
*field(integer,node4.id,10)
*string(" ")
*field(integer,node5.id,10)
*string(" ")
*field(integer,node6.id,10)
*string(" ")
*field(integer,node7.id,10)
*string(" ")
*field(integer,node8.id,10)
*string(" ")
*field(integer,0,10)
*string(" ")
*end()
*output()


*elements(103,0,"Tria3","")
*before()
*sortelements(byid)
*end()
*format()
*field(integer,id,10)
*string("")
*field(integer,propertyid,5)
*string("  ")
*field(integer,node1.id,10)
*string(" ")
*field(integer,node2.id,10)
*string(" ")
```

```
*field(integer,node3.id,10)
*string(" ")
*field(integer,0,10)
*string(" ")
*field(integer,0,10)
*string(" ")
*field(integer,0,10)
*string(" ")
*field(integer,0,10)
*string(" ")
*field(integer,0,10)
*string(" ")
*field(integer,0,10)
*string(" ")
*end()
*output()

*elements(106,0,"Tria6","")
*before()
*sortelements(byid)
*end()
*format()
*field(integer,id,10)
*string("")
*field(integer,propertyid,5)
*string("  ")
*field(integer,node1.id,10)
*string(" ")
*field(integer,node2.id,10)
*string(" ")
*field(integer,node3.id,10)
*string(" ")
*field(integer,node4.id,10)
*string(" ")
*field(integer,node5.id,10)
```

```
*string(" ")
*field(integer,node6.id,10)
*string(" ")
*field(integer,0,10)
*string(" ")
*field(integer,0,10)
*string(" ")
*field(integer,0,10)
*string(" ")
*end()
*output()

*elements(204,0,"Tetra4","")
*before()
*sortelements(byid)
*end()
*format()
*field(integer,id,10)
*string("")
*field(integer,propertyid,5)
*string("  ")
*field(integer,node1.id,5)
*string(" ")
*field(integer,node2.id,5)
*string(" ")
*field(integer,node3.id,5)
*string(" ")
*field(integer,node4.id,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
```

```
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")

*end()
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*end()


*output()
```

```
*elements(210,0,"Tetra10","")
*before()
*sortelements(byid)
*end()
*format()
*field(integer,id,5)
*string("")
*field(integer,propertyid,5)
*string("  ")
*field(integer,node1.id,5)
*string(" ")
*field(integer,node2.id,5)
*string(" ")
*field(integer,node3.id,5)
*string(" ")
*field(integer,node4.id,5)
*string(" ")
*field(integer,node5.id,5)
*string(" ")
*field(integer,node6.id,5)
*string(" ")
*field(integer,node7.id,5)
*string(" ")
*field(integer,node8.id,5)
*string(" ")
*field(integer,node9.id,5)
*string(" ")
*field(integer,node10.id,5)
*end()
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
```

```
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*end()
*output()



*elements(208,0,"Hex8","")
*before()
*sortelements(byid)
*end()
*format()
*field(integer,id,5)
*string("")
*field(integer,propertyid,5)
*string("  ")
*field(integer,node1.id,5)
*string(" ")
*field(integer,node2.id,5)
*string(" ")
*field(integer,node3.id,5)
*string(" ")
```

```
*field(integer,node4.id,5)
*string(" ")
*field(integer,node5.id,5)
*string(" ")
*field(integer,node6.id,5)
*string(" ")
*field(integer,node7.id,5)
*string(" ")
*field(integer,node8.id,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*end()
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
```

```
*string(" ")
*end()
*output()

*elements(220,0,"Hex20","")
*before()
*sortelements(byid)
*end()
*format()
*field(integer,id,5)
*string("")
*field(integer,propertyid,5)
*string("  ")
*field(integer,node1.id,5)
*string(" ")
*field(integer,node2.id,5)
*string(" ")
*field(integer,node3.id,5)
*string(" ")
*field(integer,node4.id,5)
*string(" ")
*field(integer,node5.id,5)
*string(" ")
*field(integer,node6.id,5)
*string(" ")
*field(integer,node7.id,5)
*string(" ")
*field(integer,node8.id,5)
*string(" ")
*field(integer,node9.id,5)
*string(" ")
*field(integer,node10.id,5)
*end()
*string(" ")
*field(integer,node11.id,5)
```

```
*string(" ")
*field(integer,node12.id,5)
*string(" ")
*field(integer,node13.id,5)
*string(" ")
*field(integer,node14.id,5)
*string(" ")
*field(integer,node15.id,5)
*string(" ")
*field(integer,node16.id,5)
*string(" ")
*field(integer,node17.id,5)
*string(" ")
*field(integer,node18.id,5)
*string(" ")
*field(integer,node19.id,5)
*string(" ")
*field(integer,node20.id,5)
*string(" ")
*end()
*output()

*elements(55,1,"CERIG","")
*format()
*treataslocal(counter11)
*counterset(counter11,1)
*loopif([counter11 <= dependentnodesmax])
*string("  ")
*fieldright(integer,independentnode.id,0)
*pointerset(pointer1,dependentnodes,[counter11-1])
*string("  ")
*fieldright(integer,pointer1.node.id,0)
*string(" ")
*field(integer,1,5)
*string(" ")
```

```
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")
*field(integer,0,5)
*string(" ")



*end()
*counterinc(counter11)
*endloop()
*end()
```

```
*output()



*materials()
*format()
*string("        ")
*fieldleft(integer,id,8)
*string(" ")
*treataslocal(counter16)
*counterset(counter16,1)
*if([@attributearrayvalue($MP_EY_VAL2,1) < 0.000001])
*counterset(counter16,0)
*endif()
*fieldleft(integer,counter16,8)
*end()

*treataslocal(counter1)
*counterset(counter1,1)
*loopif([counter1 <= counter10])

*if([counter16 == 0])
*string("   ")
*fieldleft(real,[@attributearrayvalue($MP_EX_VAL2,counter1)],10)
*string("    ")
*fieldleft(real,[@attributearrayvalue($MP_PRXY_VAL2,counter1)],10)
*end()
*if([counter14 == 1])
*string("    ")
*fieldleft(real,[@attributearrayvalue($MP_ALPX_VAL2,counter1)],10)
*string("   ")
*fieldleft(real,[@attributearrayvalue($MP_KXX_VAL2,counter1)],10)
*end()
*endif()
```

```
*else()
   *if([counter16 == 1])
   *string("    ")
   *fieldleft(real,[@attributearrayvalue($MP_EX_VAL2,counter1)],10)
   *string("   ")
   *fieldleft(real,[@attributearrayvalue($MP_EY_VAL2,counter1)],10)
   *string("   ")
   *fieldleft(real,[@attributearrayvalue($MP_EZ_VAL2,counter1)],10)
   *end()
   *string("    ")
   *fieldleft(real,[@attributearrayvalue(MP_GXY_VAL2,counter1)],10)
   *string("   ")
   *fieldleft(real,[@attributearrayvalue(MP_GXZ_VAL2,counter1)],10)
   *string("   ")
   *fieldleft(real,[@attributearrayvalue(MP_GYZ_VAL2,counter1)],10)
   *end()
   *string("    ")
   *fieldleft(real,[@attributearrayvalue($MP_PRXY_VAL2,counter1)],10)
   *string("   ")
   *fieldleft(real,[@attributearrayvalue($MP_PRXZ_VAL2,counter1)],10)
   *string("   ")
   *fieldleft(real,[@attributearrayvalue($MP_PRYZ_VAL2,counter1)],10)
   *end()

*if([counter14 == 1])
*string("    ")
*fieldleft(real,[@attributearrayvalue($MP_ALPX_VAL2,counter1)],10)
*string("   ")
*fieldleft(real,[@attributearrayvalue($MP_ALPY_VAL2,counter1)],10)
*string("   ")
*fieldleft(real,[@attributearrayvalue($MP_ALPZ_VAL2,counter1)],10)
*string("   ")
*fieldleft(real,[@attributearrayvalue($MP_KXX_VAL2,counter1)],10)
*end()
*endif()
```

```
   *endif()
*endif()
*string("   ")
*fieldleft(real,[@attributearrayvalue($MP_DENS_VAL2,counter1)],10)
*string("   ")
*fieldleft(real,[@attributearrayvalue($MPT_VAL2,counter1)],10)
*end()
*end()
*counterinc(counter1)
*endloop()
*output()



*text()
*string("   ")
*field(real,pointer2.pointervalue,15)
*end()
*end()
*output()


*metadata()
*format()
*if([counter15 == 1 && counter14 != 2])
*string("   ")
*pointerset(pointer1,data,6)
*field(real,pointer1.pointervalue,15)
*string("   ")
*pointerset(pointer2,data,7)
*field(real,pointer2.pointervalue,15)
*string("   ")
*pointerset(pointer3,data,8)
*field(real,pointer3.pointervalue,15)
```

```
*end()
*string("    ")
*pointerset(pointer4,data,9)
*field(real,pointer4.pointervalue,15)
*string("    ")
*pointerset(pointer5,data,10)
*field(real,pointer5.pointervalue,15)
*string("    ")
*pointerset(pointer6,data,11)
*field(real,pointer6.pointervalue,15)
*end()
*string("    ")
*pointerset(pointer7,data,12)
*field(real,pointer7.pointervalue,15)
*string("    ")
*pointerset(pointer8,data,13)
*field(real,pointer8.pointervalue,15)
*string("    ")
*pointerset(pointer9,data,14)
*field(real,pointer9.pointervalue,15)
*end()
*string("    ")
*pointerset(pointer1,data,15)
*field(real,pointer1.pointervalue,15)
*string("    ")
*pointerset(pointer2,data,16)
*field(real,pointer2.pointervalue,15)
*string("    ")
*pointerset(pointer3,data,17)
*field(real,pointer3.pointervalue,15)
*end()

*if([counter14 == 1])
*string("    ")
*pointerset(pointer4,data,18)
```

```
*field(real,pointer4.pointervalue,15)
*end()
*endif()
*endif()

*if([counter15 == 1 && counter14 == 2])

*string("   ")
*pointerset(pointer1,data,18)
*field(real,pointer1.pointervalue,15)
*end()
*string("   ")
*pointerset(pointer2,data,19)
*field(real,pointer2.pointervalue,15)
*string("   ")
*pointerset(pointer3,data,20)
*field(real,pointer3.pointervalue,15)
*string("   ")
*pointerset(pointer4,data,21)
*field(real,pointer4.pointervalue,15)
*end()
*endif()
*output()
```

Vita

Chong Teng

About the Author

Chong Teng was born in Qingdao, Shandong Province, China, at October 14, 1983. He received his B.S. degree from Nanjing University of Technology majoring in Process Material Furnishment and Control Engineering on 2006. Then he went abroad to The United States of America and finished his M.S. degree in Mechanical and Aerospace Engineering from State University of New York at Buffalo on 2008. After graduation, he moved to Utah State University to pursue a Ph.D. degree in Mechanical and Aerospace Engineering under supervision of Dr. Wenbin Yu who joined School of Aeronautics and Astronautics of Purdue University on 2013. He feels grateful about Dr. Wenbin Yu's way of guidance and enjoys the peaceful atmosphere of Logan, Utah during his Ph.D. study.

Published Journal Articles

- Teng, C., Yu, W. and Chen MY., "Variational Asymptotic Homogenization of Temperature-Dependent Heterogeneous Materials under Finite Temperature Changes," *International Journal of Solids and Structures*, Vol. 49, No. 18, 2012, pp. 2439-2449.

Journal Article in Preparation

- Teng, C. and Yu, W., "Variational Asymptotic Homogenization of Heterogeneous Materials under Nonuniformly Distributed Temperature Fields," submitting to *International Journal of Solids and Structures*, 2013

- Teng, C. and Yu, W., "Thermomechanical Micromechanics Modeling of Heterogeneous Materials with Load Effects," submitting to *International Journal of Solids and Structures*, 2013

Conference Publications

- Teng, C. and Yu,W., "Micromechanics Modeling of High Temperature Materials with Temperature-Dependent Constituents," *53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Honolulu, Hawaii, April, 2012

- Teng, C. and Yu,W., "Micromechanics Modeling of Heterogeneous Materials with Nonuniformly Distributed Loads and Temperature," *54th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Boston, Massachusetts, April, 2013

- Teng, C. and Yu,W., "Validation of Variational Asymptotic Micromechanics Modeling of Thermomechanical Behavior of Materials," *SciTech 2014*, National Harbor, Maryland, 2014 (submitted on December, 2013)